

19 JUN 2005

(19) 世界知的所有権機関
国際事務局(43) 国際公開日
2004 年 8 月 5 日 (05.08.2004)

PCT

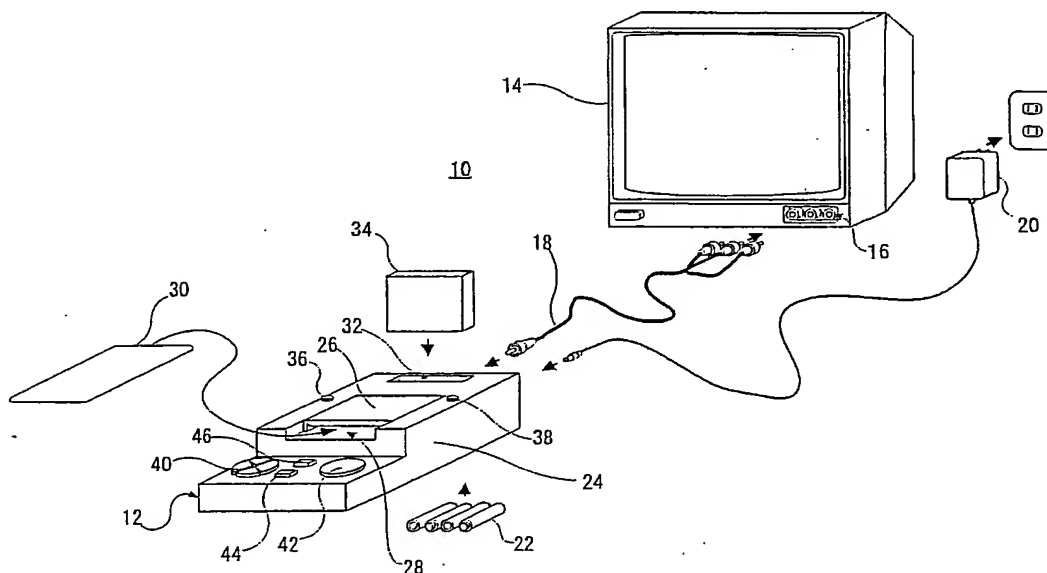
(10) 国際公開番号
WO 2004/066214 A1

- (51) 国際特許分類⁷: G06T 7/00, A63F 13/00, 1/00 (72) 発明者; および
(75) 発明者/出願人 (米国についてのみ): 上島 拓 (UESHIMA, Hiromu) [JP/JP]; 〒5250054 滋賀県草津市東矢倉 3-3-4 新世代株式会社内 Shiga (JP). 加藤 周平 (KATO, Shuhei) [JP/JP]; 〒5250054 滋賀県草津市東矢倉 3-3-4 新世代株式会社内 Shiga (JP). キム マーク カンヒョン (KIM, Mark Kanghyeon) [US/US]; 945965811 カリフォルニア州ウォルナットクリーク マイナーコート 6 California (US).
- (21) 国際出願番号: PCT/JP2004/000441
- (22) 国際出願日: 2004 年 1 月 20 日 (20.01.2004)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (30) 優先権データ: 特願2003-011644 2003 年 1 月 20 日 (20.01.2003) JP (74) 代理人: 山田 義人 (YAMADA, Yoshito); 〒5410044 大阪府大阪市中央区伏見町 2-6-6 タナベビル Osaka (JP).
- (71) 出願人 (米国を除く全ての指定国について): 新世代株式会社 (SSD COMPANY LIMITED) [JP/JP]; 〒5250054 滋賀県草津市東矢倉 3-3-4 Shiga (JP). (81) 指定国 (表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM,

[続葉有]

(54) Title: ENTERTAINMENT DEVICE USING CARDS

(54) 発明の名称: カードを用いるエンターテインメント装置



(57) Abstract: A card game device (12) includes an image sensor (54). A printed pattern (31: Fig. 2) of a card (30) is imaged by the image sensor. An imaging signal from the image sensor is sampled by a game processor (64) and further re-sampled to form an imaging pixel data arrangement. An ROM (66) has a database (67A: Fig. 7) containing a comparison data arrangement and a card ID which are set for each entry. The processor (64) searches the database according to the imaging pixel data arrangement so as to acquire one card ID and displays the card data specified by the card ID on a television monitor.

(57) 要約: カードゲーム装置 12 はイメージセンサ 54 を含み、このイメージセンサによってカード 30 の印刷図柄 (31: 図 2) を撮影する。イメージセンサからの撮影信号がゲームプロセッサ 64 によってサンプリング

[続葉有]

WO 2004/066214 A1



DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

- (84) 指定国(表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ (AT, BE, BG, CH,

CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

添付公開書類:

— 国際調査報告書

2文字コード及び他の略語については、定期発行される各PCTガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

され、さらにリサンプリングされて撮影画素データ配列が形成される。ROM 66には、比較用データ配列とカードIDとが各エントリ毎に設定されたデータベース(67A:図7)が設けられる。プロセサ64は、撮影画素データ配列に基づいてデータベースを検索することによって、1つのカードIDを取得し、そのカードIDで特定されるカードのカードデータをテレビジョンモニタ上で表示する。

明細書

カードを用いるエンターテインメント装置

技術分野

この発明はカードを用いるエンターテインメント装置に関し、特にたとえばカードに印刷された図柄を撮影し、その撮影画像に基づいてカード識別用データベースを検索して当該カードを特定し、その結果をゲーム等の入力として用いる、カードを用いるエンターテインメント装置に関する。

従来技術

従来、たとえば特許文献1（特開2001-334012号公報）、特許文献2（特開2002-136766号公報）、特許文献3（特開2002-301264号公報）、特許文献4（特開2002-224443号公報）によって、種々のカードゲームシステムが提案されている。

特許文献1には、携帯ゲーム機用のカード読取カートリッジおよび識別コードが印刷されたカードが開示されている。特許文献2には、カード読取装置と接続されたゲーム機によって、インターネットのウェブサイトにアクセスして読み取ったカードの真贋判定を行うことが開示されている。特許文献3では、業務用の多人数プレイ大型ゲーム機で、たとえばサッカー選手を印刷したカードを用いてサッカーゲームを行うものが開示されている。そして、特許文献4では紫外線インクで印刷されたバーコード等の肉眼では識別困難な印刷が施されたカードを用いるカードゲームシステムが提案されている。

従来技術に示されるように、ICカードや磁気カードのように、カードを識別するための特別な仕組みを持たない紙のカードなどの識別を行う場合、1次元/2次元バーコード等の識別用コードを印刷する必要がある。換言すれば、上記従来技術に基づくカード読み取り装置では、識別可能な専用のコードが予め記録されていないカードを識別できない。そのため、たとえばカードを識別した結果を入力とするカードゲームをプレイする場合には、それぞれ専用のカード読み取り装置とカードとが必要である。

発明の概要

それゆえに、この発明の主たる目的は、新規な、カードを用いるエンターテインメント装置を提供することである。

この発明の他の目的は、識別用コード等のないカードでも識別でき、その識別結果に基づいて入力を得ることができる、カードを用いるエンターテインメント装置を提供することである。

請求項1の発明は、それぞれに人間が視覚的に識別可能な図柄が印刷されている複数のカードから入力を得て、その入力に応じた情報処理を行うエンターテインメント装置であって、カードの図柄を撮影して撮影画素データ配列を得る撮影手段、複数のカードの各々に個別に対応する複数のエントリを含み、各々のエントリは1組のカードIDおよび比較用データ配列を含むデータベース、データベースから前記撮影画素データ配列に基づいて特定の比較用データ配列を検索し、その特定の比較用データ配列と組になっているカードIDを取得するカード識別手段、およびカード識別手段によって取得されたカードIDを入力として情報処理を実行する情報処理手段を備える、カードを用い

るエンターテインメント装置である。

請求項2の発明は、請求項1に従属し、撮影手段は図柄を撮影して撮影信号を出力するイメージセンサ、撮影信号を第1解像度でサンプリングしてデータ配列を形成するデータ配列形成手段、およびデータ配列を第1解像度より低い第2解像度でリサンプリングして撮影画素データ配列を形成する撮影画素データ配列形成手段を含み、比較用データ配列は第2解像度に対応する比較用データを含む、エンターテインメント装置である。

請求項3の発明は、請求項2に従属し、カード識別手段は、撮影画素データ配列および比較用データ配列の距離を算出し、その距離の最も小さい比較用データ配列のエントリのカードIDを取得する、エンターテインメント装置である。

請求項4の発明は、請求項3に従属し、距離は、撮影画素データ配列の各要素と、各要素に対応する比較用データ配列の各要素との差分の絶対値の総和である、エンターテインメント装置である。

請求項5の発明は、請求項3に従属し、距離は、撮影画素データ配列の各要素と、各要素に対応する比較用データ配列の各要素との差分の2乗の総和である、エンターテインメント装置である。

請求項6の発明は、請求項2ないし5のいずれかに従属し、撮影画素データ配列形成手段は、データ配列の各要素に対して所定の重み付けを行って撮影画素データ配列を形成する、エンターテインメント装置である。

請求項7の発明は、請求項2ないし6のいずれかに従属し、カード識別手段は、差分総和が所定の閾値より大きいかどうか判断する閾値判断手段を含み、差分総和が所定の閾値より大きいエントリは識別候補から排除する、エンターテインメント装置である。

請求項8の発明は、請求項7に従属し、カード識別手段は、閾値判断手段の判断の結果残った候補の総数を判断する候補数判断手段を含み、候補数判断手段が候補数「0」を判断したとき、カードIDを取得せず、候補数「1」を判断したとき、その識別候補のカードIDを取得する、エンターテインメント装置である。

請求項9の発明は、請求項8に従属し、データベースを第1データベースとし、1以上のエントリを含み、各々のエントリは複数の候補カードIDおよび1つの判定カードIDを含む第2データベースをさらに備え、カード識別手段は、識別候補が2以上残ったかどうか判断する候補数判断手段を含み、さらに候補数判断手段が候補数「2以上」を判断したとき、残った候補のカードIDの組み合わせと候補カードIDの組み合わせが一致する第2データベースのエントリを検索し、一致するエントリが存在する場合に当該エントリの判定カードIDを取得する、エンターテインメント装置である。

請求項10の発明は、請求項1ないし9のいずれかに従属し、データベースは各エントリ毎にそれに対応するカードデータを含み、さらに情報処理手段はカード識別手段によって取得されたカードIDに対応するエントリのカードデータに基づいて少なくとも図柄を表示するカードデータ表示手段を含む、エンターテインメント装置である。

請求項11の発明は、請求項1ないし10のいずれかに従属し、カートリッジコネクタをさらに備え、カートリッジコネクタにはメモリカートリッジが装着され、そのメモリカートリッジには別のデータベースが格納される、エンターテインメント装置である。

請求項12の発明は、それぞれに人間が視覚的に識別可能な図柄が印刷されている複数のカードから入力を得て、その入力に応じた情報処理を行うエンターテインメント装置であって、カードの図柄を撮影して撮影画素データ配列を得る撮影手段、撮影画素デ

ータ配列から図柄に対応するデータ列を取得するカード識別手段、およびカード識別手段によって取得されたデータ列を入力として情報処理を実行する情報処理手段を備える、カードを用いるエンターテインメント装置である。

請求項13の発明は、請求項1ないし12のいずれかに従属し、カードを所定位置にセットするカード撮影部、およびカード撮影部にセットされたカードの被撮影面に光を照射するための光源をさらに備える、エンターテインメント装置である。

請求項14の発明は、請求項12に従属し、光源からの光を拡散反射させて被撮影面に入射させる反射手段をさらに備える、エンターテインメント装置である。

請求項15の発明は、請求項13または14に従属し、カード撮影部を覆い、撮影手段と対向する面に位置補正用マークを備える撮影部カバー、および位置補正用マークに基づいて撮影画素データの取得領域を補正する撮影画素データ取得領域補正手段をさらに備え、撮像手段は、カード撮影部にカードがセットされていない状態では、位置補正用マークを撮影する、エンターテインメント装置である。

請求項16は、それぞれに人間が視覚的に識別可能な図柄が印刷されている複数のカードを撮影することによってカードを特定する方法であって、(a) 複数のカードの各々に個別に対応する複数のエントリを含み、各々のエントリは1組のカードIDおよび比較用データ配列を含むデータベースを準備し、(b) イメージセンサによって図柄を撮影して撮影信号を得て、(c) 撮影信号を第1解像度でサンプリングしてデータ配列を形成し、(d) データ配列を第1解像度より低い第2解像度でリサンプリングして撮影画素データ配列を形成し、そして(e) データベースから撮影画素データ配列に基づいて特定の比較用データ配列を検索し、その特定の比較用データ配列と組になっているカードIDを取得するようにした、カード特定方法である。

請求項17は、請求項16に従属し、ステップ(e) では、撮影画素データ配列および比較用データ配列の距離を算出し、その距離の最も小さい比較用データ配列のエントリのカードIDを取得する、カード特定方法である。

請求項1の発明では、複数のカード(30: 実施例において相当する要素またはコンポーネントの参照符号。以下、同様。)を用いる。各カードはそれぞれに人間が視覚的に識別可能なように印刷された図柄を含む。エンターテインメント装置は、そのカードの図柄を撮影することによってそのカードがどのカードか特定し、特定されたカードのカードIDを入力として、たとえばプロセサ(64)によって、情報処理を行う。具体的には、撮影手段(64, S139, S147, S149)はイメージセンサ(54)を含み、このイメージセンサからの撮影信号から撮影画素データ配列を得る。他方、データベース(67A)には、複数のカードの各々に個別に対応する複数のエントリ毎に1組のカードIDおよび比較用データ配列が設けられ、カード識別手段(64, S151)は、そのようなデータベースから撮影画素データ配列に基づいて特定の比較用データ配列を検索し、その特定の比較用データ配列と組になっているカードIDを出力する。そして、プロセサ(64)のような情報処理手段は、そのようにして得られたカードIDを入力として情報処理、たとえばコンピュータゲーム処理を実行する。

請求項2では、撮影手段は図柄を撮影して撮影信号出力するイメージセンサ(54)を含み、この撮影信号がデータ配列形成手段(S77)に与えられ、そのデータ配列形成手段が撮影信号を第1解像度でサンプリングしてデータ配列を形成する。実施例ではリサンプリング手段である撮影画素データ配列形成手段(S147)は、第1解像度の

データ配列をそれより低い第2解像度でリサンプリングして撮影画素データ配列を形成する。ただし、必要に応じて、直流成分除去および正規化処理を施して影画素データ配列を形成してもよい。

請求項2の発明では、第2解像度でリサンプリングして撮影画素データ配列を形成するようにしているため、必要な識別精度を保持しながら、識別のための処理負荷とデータベースサイズを小さくすることができる。

請求項3の発明では、カード識別手段は、撮影画素データ配列および比較用データ配列の距離を算出し、その距離の最も小さい比較用データ配列のエントリのカードIDを取得する。

請求項4の発明では、距離は、撮影画素データ配列の各要素と、各要素に対応する比較用データ配列の各要素との差分の絶対値の総和である。

請求項5の発明では、距離は、撮影画素データ配列の各要素と、各要素に対応する比較用データ配列の各要素との差分の2乗の総和である。

請求項3-5の発明によれば、データベースを参照して行うカード識別が容易である。

請求項6では、撮影画素データ配列形成手段は、たとえば2次元窓関数(図26)を用いて、データ配列の各要素に対して所定の重み付けを行って撮影画素データ配列を形成する。

請求項6によれば、リサンプリング時に中心に近いピクセルの重みを大きくし、かつ中心から離れたピクセルの重みを小さくすることによって、中心に近いピクセルをより反映したりサンプリングピクセル値を得ることができる。

請求項7では、閾値判断手段(S183)において差分総和が所定の閾値より大きいと判断したエントリは識別候補から排除される(S189)。

請求項8では、第1候補数判断手段(S193)が候補数「0」を判断したとき、「一致カードなし」となりカードIDを取得しない(S195)。さらに、第2候補数判断手段(S197)が候補数「1」を判断したとき、その識別候補のカードIDを取得する(S199)。

請求項9では、第2データベース(67B)をさらに設け、候補数判断手段(S197)が候補数「2以上」を判断したとき、たとえばステップS217で、残った候補のカードIDの組み合わせと候補カードIDの組み合わせが一致する第2データベースのエントリを検索し、一致するエントリが存在する場合に当該エントリの判定カードIDを取得する。

請求項5-9の具体的識別手法によれば、確実に1つのカードが特定できる。

請求項10では、カードデータ表示手段(S127;100)が、データベース(67A)から読み出したカードデータによって、図柄やその他必要なカード情報を表示する。

請求項10によれば、プレイヤーが提示したカードがどのように識別または認識されたカードが確認でき、したがって、プレイヤーにエンターテインメント装置の信頼性を与えることができる。

請求項11では、メモ리카ートリッジ(34)がカートリッジコネクタ(68)に挿入されると、そのカートリッジコネクタを通してメモ리카ートリッジがプロセサに結合される。したがって、メモ리카ートリッジを用いることにより、データベースの追加、更新、交換が可能となる。さらに、このメモ리카ートリッジを用いることによって、ソ

ソフトウェアの追加、更新あるいは交換を行うこともできる。

請求項12の発明は、複数のカード(30)を用いる。各カードはそれぞれに人間が視覚的に識別可能なように印刷された図柄を含む。エンターテインメント装置は、撮影手段を含み、この撮影手段(64, S139, S147, S149)はイメージセンサ(54)を含み、このイメージセンサからの撮影信号から撮影画素データ配列を得る。カード識別手段は、この撮影画素データ配列を処理することによってデータ列を取得し、プロセサ(64)のような情報処理手段は、そのようにして得られたデータ列を入力として情報処理を実行する。

請求項13では、カード撮影部(28)にセットされたカードの被撮影面に、光源すなわちLED(58)からの光が照射される。

そして、請求項14のように、反射手段(60)を用いれば、カードの被撮影面を均一に照射することができる。

請求項15では、カード撮影部カバー(26)の撮影手段と対向する面すなわち裏面(27)に位置補正用マーク(62)を設け、撮像手段によってカード撮影部にカードがセットされていない状態で位置補正用マークを撮影することによって、撮影画素データ取得領域補正手段(S21, S23)が撮影画素データの取得領域を補正する。

この請求項15によれば、イメージセンサ等に位置ずれが生じていても、確実にカードの図柄を撮影することができる。

請求項16では、イメージセンサによって図柄を撮影してえられた撮影信号を第1解像度でサンプリングしてデータ配列を形成し、そのデータ配列を第1解像度より低い第2解像度でリサンプリングして撮影画素データ配列を形成し、データベースからその撮影画素データ配列に基づいて特定の比較用データ配列を検索し、その特定の比較用データ配列と組になっているカードIDを取得する。

請求項17では、カード識別ステップ(S151, S181)では、撮影画素データ配列および比較用データ配列の距離(たとえば、ユークリッド距離、ハミング距離、など)を算出し、その距離の最も小さい比較用データ配列のエントリのカードIDを取得する。

この発明によれば、図柄さえ印刷されていればそのカードを識別することができるので、既に市場に出回っている識別コード等を持たないカードを利用してカードゲーム等のエンターテインメントを楽しむことができる。

この発明の上述の目的、その他の目的、特徴、および利点は、図面を参照して行う以下の実施例の詳細な説明から一層明らかとなる。

図面の簡単な説明

図1はこの発明の一実施例のカードゲームシステムの全体構成を示す図解図である。

図2は図1実施例に用いられるカードの一例を示す図解図である。

図3は図1実施例のカードゲーム機を上からみた図解図である。

図4は図1実施例のカードゲーム機の図2の線III-IIIからみた断面図解図である。

図5は図1実施例のカードゲーム機のカード撮影部カバーの裏面を示す図解図である。

図6は図1実施例を示すブロック図である。

図7はデータベースAの一例を示す図解図である。

図8はデータベースBの一例を示す図解図である。

図9は図6実施例においてイメージセンサからゲームプロセッサへピクセルデータを取り込む構成およびLED駆動回路を示す回路図である。

図10は図9 6実施例の動作を示すタイミング図である。

図11は図10の一部を拡大して示すタイミング図である。

図12は図1実施例の全体動作を示すフロー図である。

図13は図12実施例のシステム初期化動作を示すフロー図である。

図14は図13実施例のシステム初期化処理におけるセンサ初期化動作を示すフロー図である。

図15は図14のコマンド送信処理動作を示すフロー図である。

図16は図14のレジスタ設定処理動作を示すフロー図である。

図17は図16に示すレジスタ設定処理動作を示すタイミング図である。

図18は図13実施例のシステム初期化処理における撮影動作を示すフロー図である。

図19は図18実施例の撮影動作における画素データ配列取得処理動作を示すフロー図である。

図20は図19の画素データ配列取得処理における画素データ取得処理動作を示すフロー図である。

図21は図20の撮影動作によって撮影された撮影部カバーの裏面のマークの画像を示す図解図である。

図22はマークから中心位置を求めその周囲に有効画素領域を設定することを示す図解図である。

図23は図12実施例におけるゲームプレイ動作を示すフロー図である。

図24は図23実施例におけるカード撮影および識別動作を示すフロー図である。

図25は図24実施例におけるリサンプリング動作を示すフロー図である。

図26は図25実施例におけるリサンプリングの方法を示す図解図である。

図27は図24実施例における直流成分除去および正規化動作を示すフロー図である。

図28は図24実施例におけるデータベース検索動作を示すフロー図である。

図29はデータベース検索動作の続きを示すフロー図である。

図30は図1実施例においてテレビジョンモニタ上に表示されるカード画面の一例を示す図解図である。

図31は図1実施例においてテレビジョンモニタ上に表示されるゲーム画面の一例を示す図解図である。

発明を実施するための最良の形態

図1を参照して、この発明の一実施例であるカードゲームシステム10は、カードゲーム機(以下、単に「ゲーム機」と呼ばれることもある。)12およびたとえば家庭用テレビジョン受像機のようなテレビジョンモニタ14を含む。このゲーム機12は、AVケーブル18を通して、テレビジョンモニタ14のAV端子16に接続される。そして、ゲーム機12には、ACアダプタ20によって直流電源が与えられる。ただし、それは電池22に代えられてもよい。

ゲーム機12は、また、ハウジング24を含み、このハウジング24の上面は階段状に形成され、上段および下段を有する。ハウジング24の上段上面にはカード撮影部カバー26が設けられる。このカバー26の下面とハウジング24の上面との間に隙間が

形成さ、この隙間がカード挿入部 28 となる。つまり、このカード挿入部 28 に、具体的には図 2 に示すようなカード 30 が挿入され、このカード 30 の図柄表示部 31 (図 2) が後述のようにして撮影される。

ハウジング 24 の上段のカバー 26 より奥側の上面にはカートリッジ挿入口 32 が形成され、このカートリッジ挿入口 32 に、カートリッジ 34 が挿入される。後述するように、このカートリッジ 34 が、コネクタを通して、ゲームプロセサと電氣的に結合される。ハウジング 24 の上段のカバー 26 を挟む両側には、電源スイッチ 36 およびリセットスイッチ 38 が設けられる。

ハウジング 24 の下段上面左右両側には、方向キー 40 および決定およびカード撮影キー 42 が設けられる。方向キー 40 は、4 方向 (上下左右) の個別のボタンを有し、たとえばテレビジョンモニタ 14 の表示画面上においてメニューやゲームモード選択のためにカーソルを移動させたりするために用いられる。決定およびカード撮影キー 40 はゲーム機 12 への入力を決定するために用いられるとともに、カード挿入口 26 に挿入したカードを撮影するときに押される。また、2 つのボタン 40 および 42 に挟まれて、ハウジング 24 の下段上面には、キャンセルキー 44 およびポーズキー 46 が設けられる。キャンセルキー 44 は、ゲーム機 12 への入力をキャンセルするために用いられる。ポーズキー 46 は一時的に停止するために使用される。

図 3 および図 4 には図 1 実施例のカードゲーム機 12 がより詳細に図解される。すなわち、ゲーム機 12 は上述のようにハウジング 24 を有し、このハウジング 24 はそれぞれプラスチックの射出成型で形成された上ハウジング 24 a および下ハウジング 24 b で構成され、この上下ハウジング 24 a および 24 b は上下ハウジング接合用ボス 48 によって接合または連結され、1 つのハウジング 24 となる。上ハウジング 24 a にはその天井から下垂するように基板ボス 50 が設けられる。この基板ボス 50 の先端 (下端) にはビスによってプリント基板 52 が取り付けられ、このプリント基板 52 には、イメージセンサ 54 やその関連の素子および後述のゲームプロセサ 64 (図 6) などの電子部品が装着され、必要な回路を構成する。

イメージセンサ 54 は、詳細には図示してはいないが、たとえばプラスチック射出成型によってそれぞれ形成されるユニットベースとそのユニットベースに指示される支持筒とを含む。図 4 からよくわかるように、この支持筒の上面には内面が逆円錐形状であるラッパ状の開口が形成される。そして、その開口の下方の筒状部内部には、いずれもがたとえば透明プラスチックの成型によって形成された凹レンズおよび凸レンズを含む光学系が設けられ、凸レンズの下方に撮像素子が固着される。したがって、イメージセンサ 54 は、支持筒の上部開口からレンズ系を通して入射する光に応じた映像を撮影することができる。具体的な構造は、同じ出願人の出願に係る同時係属中の特願 2002-346052 号の図 2 を参照されたい。なお、この実施例では、イメージセンサ 54 は、低解像度の CMOS イメージセンサ (たとえば 32×32 画素: グレースケール) を用いる。ただし、このイメージセンサ 54 の撮像素子の画素数はもっと多くされてもよいし、CCD 等の他の撮像素子を用いるものであってよい。

上ハウジング 24 a の天井であってカバー 26 の下方には透明樹脂板 56 が嵌め込まれ、カード挿入部 28 (図 3) から挿入されたカード 30 (図 2) がその上に載る。このカード 30 の図柄 31 をイメージセンサ 54 で撮影するのであるが、カード 30 の周囲は、物に囲まれている状態なので、かなり暗く、そのままではイメージセンサ 54 に

よる撮影がうまくできない。そこで、この実施例では、上部ハウジング24a内に複数(実施例では4つ)の赤色発光ダイオード(以下、「LED」ということもある。)58を設け、そのLED58からの光がハウジング24内で反射されて透明樹脂板56を通してカード30の被撮影面を照射する。このときの光の反射を効率的に行なわせるように、基板52の上方に反射板60設ける。この反射板60は、実施例では、その表面を梨地加工した白色樹脂板である。梨地加工はLED58からの光を乱反射させて可及的均一に被撮影面全体を明るくするための工夫であり、別の粗面加工が用いられてもよい。さらに、反射板60は、プラスチックでなく、たとえば紙のような別の素材で作られてもよい。

なお、実施例では赤色LEDを用いた。しかしながら、他の色のLEDでもよく、LED以外の光源を用いてもよい。さらには、カラー撮影が必要な場合には、このような光源としては、白色光源が用いられるべきである。

なお、撮影部カバー26の内側面27(図5)も、反射を効率よく行えるように、白色プラスチック(または紙)によって形成される。したがって、カード30の被撮影面は、カバー26の内側面27から反射された光によっても明るくされる。そして、また、カバー26の内側面27上には、複数(この実施例では4つ)の黒色マーク62が形成される。この4つのマーク62は、後述のように、イメージセンサ54やその他の位置ずれ(たとえば、取り付け位置のずれや部品の寸法誤差などに起因する)を検出して、その位置ずれを補正または校正するためのものである。簡単にいうと、4つの黒色マーク62で囲まれる領域の中心を見つけ、その中心を含む一定ピクセル分を有効撮影領域として決める。それによって、個々のゲーム機で仮にイメージセンサ54等の取り付け位置や寸法がずれていても、各ゲーム機毎に有効撮影領域が規定されるので、確実にカード30の図柄31(図2)を撮影することができる。

図6を参照して、赤色発光ダイオード58からの光に照射され、カード30上に印刷されている図柄31がイメージセンサ54によって撮影される。したがって、イメージセンサ54からはカード図柄の映像信号が出力される。イメージセンサ54からのこのアナログ映像信号はゲームプロセサ64に内蔵されたA/Dコンバータ(図示せず)によってデジタルデータに変換される。

このようなゲームプロセサ64としては、任意の種類のプロセサを利用できるが、この実施例では、本件出願人が開発しかつ既に特許出願している高速プロセサを用いる。この高速プロセサは、たとえば特開平10-307790号公報[G06F13/36, 15/78] およびこれに対応するアメリカ特許第6,070,205号に詳細に開示されている。

ゲームプロセサ64は、図示しないが、演算プロセサ、グラフィックプロセサ、サウンドプロセサおよびDMAプロセサ等の各種プロセサを含むとともに、アナログ信号を取り込むときに用いられる上述のA/Dコンバータやキー操作信号や赤外線信号のような入力信号を受けかつ出力信号を外部機器に与える入出力制御回路を含む。したがって、操作キー42-46からの入力信号がこの入出力制御回路を経て、演算プロセサに与えられる。演算プロセサは、その入力信号に応じて必要な演算を実行し、その結果をグラフィックプロセサ等に与える。したがって、グラフィックプロセサやサウンドプロセサはその演算結果に応じた画像処理や音声処理を実行する。

プロセサ64には、図示しないが内部メモリが設けられ、この内部メモリは、ROM

またはRAM（SRAMおよび／またはDRAM）を含む。RAMは一時メモリ、ワーキングメモリあるいはカウンタまたはレジスタ領域（テンポラリデータ領域）およびフラグ領域として利用される。なお、ゲームプロセッサ64にはROM66が外部バスを通して接続される。このROM66に後に説明するようなゲームプログラムや、カード識別用データベースが予め設定される。

ゲームプロセッサ64は、イメージセンサ54からA/Dコンバータを介して入力されるデジタル映像信号を処理してそのときカード撮影部にあるカードがどのカードかを特定するとともに、操作キー40-46からの入力信号に従って演算、グラフィック処理、サウンド処理等を実行し、ビデオ信号およびオーディオ信号を出力する。ビデオ信号はゲーム画面を表示するための画像信号であり、オーディオ信号はゲーム音楽や効果音の信号であり、したがって、テレビジョンモニタ（図示せず）の画面上にゲーム画面表示され、必要なサウンド（効果音、ゲーム音楽）がそのスピーカから出力される。

実施例のカードゲーム機12はさらに、外部バスに接続されたカートリッジコネクタ68を有する。カートリッジコネクタ68は図1に示すカートリッジ挿入口32奥部に設けられるもので、このカートリッジコネクタ68に、必要に応じて、カートリッジ34が接続される。このカートリッジ34には外部ROM70が内蔵されていて、このROM70には、一例では、識別すべきカードの種類が変更されたときにその変更された種類のカードに対応できるように識別用データベースが格納される。したがって、この場合には、カートリッジ34は、内部ROM66とともに用いられる。

ただし、カートリッジ34を、同じ出願人の出願に係る同時係属中の特開2002-132509号と同じように、別のプログラムを起動させるために使用することもできる。その場合には、ROM70内には、カード識別用データベースだけでなく、プログラムも予め設定されている必要がある。

つまり、メモリカートリッジ34内のROM70には、ある実施例では、後述のデータベースのデータの追加、更新あるいは交換を行うためのデータが格納される。これに対して、別のプログラムを起動する実施例では、ROM70内には、上述のデータベース用データの他に、ソフトウェアの追加、更新或いは交換を行うためのデータやプログラムを格納しておく必要がある。

識別用データベースの一例が図7および図8に示される。図7の識別用データベース67Aは、有効領域（実施例では $24 \times 24 = 576$ ピクセル）を 3×3 ピクセルでリサンプリング（後述）した後のピクセル（リサンプリングピクセル）毎に比較用ピクセル値を格納しておき、そのときの撮影信号を処理しかつリサンプリングした後の各ピクセルのピクセル値（リサンプリングピクセル値）と比較する。なお、リサンプリング後も用語「ピクセル」を使うのは、実際には元々9ピクセルあったものであるから、厳密には多少違和感があるが、便宜的に、「元の9ピクセルを1つにリサンプリングした」という意味で、用語「リサンプリングピクセル」を用いる。）そして、そのときのカードの撮影データから得られるリサンプリングピクセル値とデータベース67Aに記憶した比較用ピクセル値とが一致すれば、そのカードはたとえば「エントリ#2」のカードである、というような特定が行える。

なお、図7に示すように、このデータベース67Aには、さらに、各エントリ毎に、そのエントリで特定されるカードIDで表されるカードのカードデータ（図柄、名称、特徴、パワー値等）が予め設定されていて、後に説明するように、ゲームプレイ中にカ

ードを特定したときに、そのカードデータを用いてカード画面100（図30）を表示する。

図7のデータベース67Aが各リサンプリングピクセル値を対比して1つのカードを特定するためのデータベースである。各リサンプリングピクセルのピクセル値がデータベース67Aのリファレンス値と一致すればそのとき撮影したカードがどれか1つエントリで示すカードであると特定することはできる。しかしながら、すべてのリサンプリングピクセル値が常に一致するとは限らず、いくつかのリサンプリングピクセルで異なるピクセル値になる場合には、カードの特定ができなくなってしまう。それではカードゲームが進行できないので、一致ではなくても、どれかのカードであると特定する必要がある。図8の識別用データベース67Bはそのような場合に使うデータベースである。つまり、図8のデータベース67Bは、1つのカードを特定できなくて複数の候補が最後まで残った場合に、その候補から1つのカードを特定するために使用する。

図8のデータベース67Bでは、識別カードIDとして、図7のデータベースを使って特定したエントリ番号（#）が格納されていて、たとえば最後まで残った2つの候補が、#22のカード、#4のカードであると認識された場合、それは#4のカードであると特定する。つまり、この図8のデータベース67Bは、順次の認識結果を候補パターンとして格納していて、その認識順の候補パターンが一致したものを最終的に1つのカードとして特定する。

なお、図7および図8のデータベース67Aおよび67Bは内部ROM66（図6）に格納されるものであるが、カートリッジ34を用いる場合には、当然、外部ROM70にも同様のデータベースが設定可能であることはもちろんである。

ここで、図9～図11を参照して、CMOSイメージセンサ54からゲームプロセッサ64へピクセルデータを取り込むための構成を詳細に説明する。図9に示すように、実施例のCMOSイメージセンサ54は、ピクセル信号（画素信号）をアナログ信号として出力するタイプのものであるため、このピクセル信号はゲームプロセッサ64のアナログ入力ポートに入力される。アナログ入力ポートは、このゲームプロセッサ64内においてA/Dコンバータ（図示せず）に接続され、したがって、ゲームプロセッサ64は、A/Dコンバータからデジタルデータに変換されたピクセル信号（ピクセルデータ）をその内部に取得する。

上述のアナログピクセル信号の中心は、CMOSイメージセンサ54の基準電圧端子Vrefに与えられる基準電圧によって決定される。そのため、この実施例では、イメージセンサ54に関連してたとえば抵抗分圧回路からなる基準電圧発生回路72が設けられ、この回路72から基準電圧端子Vrefに常に一定の大きさの基準電圧が与えられる。

CMOSイメージセンサ54を制御するための各デジタル信号は、ゲームプロセッサ64のI/Oポートに与えられ、またはそこから出力される。このI/Oポートは各々入力/出力の制御が可能なデジタルポートであり、このゲームプロセッサ64内で入出力制御回路（図示せず）に接続されている。

詳しくいうと、ゲームプロセッサ64の出力ポートからはイメージセンサ54をリセットするためのリセット信号が出力され、イメージセンサ54に与えられる。また、イメージセンサ54からは、ピクセルデータストローブ信号およびフレームステータスフラグ信号が出力され、それらの信号がゲームプロセッサ64の入力ポートに与えられる。ピ

クセルデータストローブ信号は上述の各ピクセル信号を読み込むための図10(b)に示すようなストローブ信号である。フレームステータスフラグ信号はイメージセンサ54の状態を示すフラグ信号で、図10(a)に示すように、このイメージセンサの露光期間を規定する。つまり、フレームステータスフラグ信号の図10(a)に示すローレベルが露光期間を示し、図10(a)に示すハイレベルが非露光期間を示す。

また、ゲームプロセッサ64は、CMOSイメージセンサ54内の制御レジスタ(図示せず)に設定するコマンド(またはコマンド+データ)をレジスタデータとしてI/Oポートから出力するとともに、たとえばハイレベルおよびローレベルを繰り返すレジスタ設定クロックを出力し、それらをイメージセンサ54に与える。

なお、この実施例では、赤色発光ダイオード58として、図9に示すように互いに並列接続された4つの赤色発光ダイオード58、58、…を用いる。この4つの赤色LED58は、上で説明したように、対象物(撮影すべきカード30)を照らすように、イメージセンサ54を囲むように配置される。このLED58はLED駆動回路74によって、点灯されまたは消灯(非点灯)される。LED駆動回路74は、イメージセンサ54から上述のフレームステータスフラグ信号を受け、このフラグ信号は、抵抗78およびコンデンサ80からなる微分回路76を通してPNPトランジスタ84のベースに与えられる。このPNPトランジスタ84にはさらにプルアップ抵抗82が接続されていて、このPNPトランジスタ84のベースは、通常は、ハイレベルにプルアップされている。そして、フレームステータス信号がローレベルになると、そのローレベルが微分回路76を経てベースに入力されるため、PNPトランジスタ84は、フラグ信号がローレベル期間にのみオンする。

PNPトランジスタ84のエミッタは抵抗86および88を介して接地される。そして、エミッタ抵抗86および88の接続点がNPNトランジスタ90のベースに接続される。このNPNトランジスタ90のコレクタが各赤色LED58のアノードに共通に接続される。NPNトランジスタ90のエミッタが別のNPNトランジスタ92のベースに直接接続される。NPNトランジスタ92のコレクタが各LED58のカソードに共通接続され、エミッタが接地される。

このLED駆動回路74では、ゲームプロセッサ64のI/Oポートから出力されるLEDコントロール信号がアクティブ(ハイレベル)でありかつイメージセンサ54からのフレームステータスフラグ信号がローレベルである期間にのみLED58が点灯される。図10(a)に示すようにフレームステータスフラグ信号がローレベルになると、そのローレベル期間中(実際には微分回路76の時定数での遅れがあるが)、PNPトランジスタ84がオンする。したがって、図10(d)に示すLEDコントロール信号がゲームプロセッサ64からハイレベルで出力されると、NPNトランジスタ90のベースがローレベルとなり、このトランジスタ90がオフとなる。トランジスタ90がオフするとトランジスタ92はオンとなる。したがって、電源(図9では小さい白丸で示す)から各LED58およびトランジスタ92を経て電流が流れ、応じて図10(e)に示すように各LED58が点灯される。

実施例のLED駆動回路74では、このように、図10(d)のLEDコントロール信号がアクティブでありかつ図10(a)のフレームステータスフラグ信号がローレベルである期間、すなわちイメージセンサ54の露光期間(図10(f)参照)にのみLED58が点灯されることになる。したがって、この実施例によれば、無駄な電力消費

を抑制することができる。さらに、フレームステータスフラグ信号はコンデンサ80によってカップリングされているので、万一イメージセンサ54の暴走等によりそのフラグ信号がローレベルのまま停止した場合でも、一定時間後にはトランジスタ84は必ずオフされ、LED58も一定時間後には必ずオフされる。

このように、この実施例では、フレームステータス信号の持続期間を変更することによって、イメージセンサ54の露光時間を任意にかつ自在に設定または変更することができる。

さらに、フレームステータス信号およびLEDコントロール信号の持続時間や周期を変更することによって、LED58の発光期間、非発光期間、発光／非発光周期などを任意にかつ自在に変更または設定できる。

先に説明したように、LED58からの赤色光によってカード30の被撮影面すなわち図柄31（図2）が照射されると、それからの反射光によってイメージセンサ54が露光される。応じて、イメージセンサ54から上述の 픽셀信号が出力される。詳しく説明すると、CMOSイメージセンサ54は、上述の図10（a）のフレームステータスフラグ信号がハイレベルの期間（LED58の非点灯期間）に図10（b）に示す 픽셀データストローブに同期して、図10（c）に示すようにアナログの 픽셀信号を出力する。ゲームプロセッサ64では、そのフレームステータスフラグ信号と 픽셀データストローブとを監視しながら、A/Dコンバータを通じて、デジタルの 픽셀データを取得する。

ただし、픽셀データ（픽셀信号）は実施例では、図11（c）に示すように、第0行、第1行、…第31行と行順次に出力される。ただし、後に説明するように、各行の先頭の1 픽셀はダミーデータとなる。

なお、上述の実施例では、必要なときにのみLED58を点灯してカード30を照射するようにした。しかしながら、電源スイッチ36（図1）をオンしているときにはLED58を常に点灯するようにしてもよい。

次に、図12以降のフロー図に基づいて図1実施例のカードゲームシステム10の動作ないし操作を説明する。図1に示す電源スイッチ36をオンしてゲームスタートとなるが、図6に示すゲームプロセッサ64は、まず、ステップS1で初期化処理を実行する。システム初期化ルーチンが図13に示され、この図13の最初のステップS11では、システム（ハードウェア）およびメモリ（各変数）を初期化する。

次いで、ステップS13で、センサ初期化処理を実行する。このセンサ初期化処理は、イメージセンサ54内の制御レジスタへのデータ設定処理を含み、具体的には、図14～図16に示すフロー図に従って、かつ図17に示すタイミングに従って実行される。

図14の最初のステップS27では、ゲームプロセッサ64は、設定データとして、コマンド“CONF”を設定する。ただし、このコマンド“CONF”は、イメージセンサ54に、ゲームプロセッサ64からコマンドを送信する設定モードに入ることを知らせるためのコマンドである。そして、次のステップS29で、図15に詳細に示すコマンド送信処理を実行する。

コマンド送信処理の最初のステップS47では、プロセッサ64は、設定データ（ステップS29の場合はコマンド“CONF”）をレジスタデータ（I/Oポート）に設定し、次のステップS49でレジスタ設定クロック（I/Oポート）をローレベルに設定する。その後、ステップS51で規定時間待機した後、ステップS53で、レジスタ設定クロ

ックをハイレベルに設定する。そして、さらにステップS 5 5での規定時間の待機の後、ステップS 5 7でレジスタ設定クロックを再びローレベルに設定する。このようにして、図1 7に示すように、規定時間の待機を行いながら、レジスタ設定クロックをローレベル、ハイレベルそしてローレベルとすることによって、コマンド（コマンドまたはコマンド+データ）の送信処理が行われる。

図1 4に戻って、ステップS 3 1では、ピクセルモードを設定するとともに、露光時間の設定を行う。この実施例の場合、イメージセンサ5 4は先に述べたようにたとえば 32×32 のCMOSセンサであるため、設定アドレス“0”のピクセルモードレジスタに 32×32 画素であることを示す“0h”を設定する。次のステップS 3 3において、ゲームプロセッサ6 4は、図1 6に詳細に示すレジスタ設定処理を実行する。

レジスタ設定処理の最初のステップS 5 9では、プロセッサ6 4は、設定データとして、コマンド“MOV”+アドレスを設定し、次のステップS 6 1で、図1 5で先に説明したコマンド送信処理を実行して、それを送信する。次にステップS 6 3において、プロセッサ6 4は、設定データとして、コマンド“LD”+データを設定し、次のステップS 6 5でコマンド送信処理を実行して、それを送信する。そして、ステップS 6 7で、プロセッサ6 4は、設定データとして、コマンド“SET”を設定し、次のステップS 6 9でそれを送信する。なお、コマンド“MOV”は制御レジスタのアドレスを送信することを示すコマンドで、コマンド“LD”はデータを送信することを示すコマンドで、コマンド“SET”はデータをそのアドレスに実際に設定させるためのコマンドである。なお、この処理は、設定する制御レジスタが複数ある場合には、繰り返し実行される。

再び図1 4に戻って、次のステップS 3 5では、設定アドレスを“1”（露光時間設定レジスタのローニブルのアドレスを示す）とし、最大露光時間を示す“FFh”のローニブルデータ“Fh”を設定すべきデータとして設定する。そして、ステップS 3 7で図1 6のレジスタ設定処理を実行する。同様に、ステップS 3 9において、設定アドレスを“2”（露光時間設定レジスタのハイニブルのアドレスを示す）とし、最大露光時間を示す“FFh”のハイニブルデータ“Fh”を設定すべきデータとして設定し、ステップS 4 1でレジスタ設定処理を実行する。

その後、ステップS 4 3で設定終了を示しかつイメージセンサ5 4にデータの出力を開始させるためのコマンド“RUN”を設定し、ステップS 4 5で送信する。このようにして、図1 3に示すステップS 1 3でのセンサ初期化動作が実行される。

図1 3の次のステップS 1 5において、図1 8に詳細に示す撮影ルーチンが実行される。図1 8を参照して、最初のステップS 7 1ではプロセッサ6 4の内部RAM（図示せず）に形成されるテンポラリデータ配列（これは $32 \times 32 = 1024$ ピクセル分の記憶場所で形成される）の全要素に「0」を設定する。つまり、テンポラリデータ配列を初期化する。その後、ステップS 7 3でイメージセンサ5 4をリセットするとともに、ステップS 7 5でLED 5 8を点灯する。そして、次のステップS 7 7で、画素（ピクセル）データ配列の取得処理を実行する。

この画素データ配列取得処理ルーチンは具体的には、図1 9で示される。図1 9の最初のステップS 7 8で、ゲームプロセッサ6 4は画素データ配列の要素番号としてXに「-1」、Yに「0」を設定する。この実施例における画素データ配列は、 $X = 0 \sim 31$ 、 $Y = 0 \sim 31$ の2次元配列であるが、前述のように各行の先頭ピクセルのデータとしてダミーデータが出力されるので、Xの初期値として「-1」が設定される。続くステップ

S 8 0では、図20に示す要素[Y][X]の画素データの取得処理を実行する。

図20の最初のステップS 9 3で、ゲームプロセサ64は、イメージセンサ54からのフレームステータスフラグ信号をチェックし、ステップS 9 5でその立ち上がりエッジ（ローレベルからハイレベルへの）が発生したかどうか判断する。そして、ステップS 9 5でフラグ信号の立ち上がりエッジを検出すると、次のステップS 9 7において、ゲームプロセサ64は、その内部のA/Dコンバータに入力されてきたアナログのピクセル信号のデジタルデータへの変換の開始を指示する。その後、ステップS 9 9でイメージセンサ54からのピクセルストローブをチェックし、ステップS 1 0 1でそのストローブ信号のローレベルからハイレベルへの立ち上がりエッジが発生したかどうか判断する。

ステップS 1 0 1で“YES”が判断されると、ゲームプロセサ64は続いて、ステップS 1 0 3において、 $X = -1$ かどうか、すなわち先頭ピクセルかどうか判断する。先に述べたように、各行の先頭ピクセルはダミーピクセルとして設定されているので、このステップS 1 0 3で“YES”が判断されると、次のステップS 1 0 5でそのときのピクセルデータを取得しないで、ステップS 1 0 7で要素番号Xをインクリメントする。

ステップS 1 0 3で“NO”が判断されると、行の第2番目以降のピクセルデータであるので、ステップS 1 0 9およびS 1 1 1において、そのときのピクセルデータを取得し、テンポラリレジスタ（図示せず）にそのピクセルデータを格納する。その後、図19のステップS 8 2にリターンする。

ステップS 8 2では、テンポラリレジスタに格納されたピクセルデータを画素データ配列の要素[Y][X]として格納する。

続くステップS 8 4でXをインクリメントする。ステップS 8 6でXが「32」に満たないと判断した場合、前述のステップS 8 0からステップS 8 4の処理を繰り返し実行する。ステップS 8 6でXが「32」に達したと判断した場合、すなわち画素データの取得が行の終端に到達した場合には、続くステップS 8 8でXに「-1」を設定し、ステップS 9 0でYをインクリメントする。ステップS 9 2でYが「32」未満であると判断した場合、ステップS 8 0に戻って、次の行の先頭から画素データの取得処理を繰り返す。ステップS 9 2でYが「32」になったと判断した場合には、すなわち画素データの取得が画素データ配列の終端に到達した場合には、図18のステップS 7 9にリターンする。

図18のステップS 7 9でLED 58を消灯する。その後、ステップS 8 1において、取得したピクセルデータの各要素をテンポラリデータ配列の各要素に加算する。この処理はローコントラストの撮影データでも確実に読み取るために実行するステップであり、テンポラリデータ配列の各要素（各ピクセル）毎に、ピクセル値を加算する。そして、ステップS 8 3において、テンポラリデータ配列の全要素（ $32 \times 32 = 1024$ ）のピクセル値の総和を求めた後その総和を要素数（実施例では、1024）で除算することによって、テンポラリデータ配列の全要素の平均ピクセル値を計算するとともに、全要素中から最小のピクセル値を検出する。

続くステップS 8 5で、「平均値-最小値」が所定の閾値より大きいかどうか判断する。つまり、コントラストが所定の閾値以上になったかどうか判断する。ここでの閾値は、それ以後誤りなく画像処理が実行できると想定したコントラスト（平均ピクセル値-最

小ピクセル値)である。ステップS 8 5で“NO”が判断されると、つまり、必要最小限のコントラストにも達していないときには、さらにステップS 8 1およびS 8 3を繰り返すべく、ステップS 8 7でカウンタをインクリメントした後、ステップS 8 9で、そのカウンタのカウント値が規定値(実施例では3 2回または6 4回とする)より大きくなったかどうか判断する。このステップS 8 9で“YES”が判断されると、コントラストは充分ではないが、ピクセルデータの読み取り回数が多く、時間がかかりすぎるので、先のステップS 8 5で“YES”が判断された場合と同様に、次のステップS 9 1を実行する。このステップS 9 1では、テンポラリデータ配列の各要素から最小値を減算するとともに、減算後に規定の最大(ピクセル)値を超えている要素を最大値に飽和させる。詳しく言うと、全ての要素の値は最小値以上の値であり、このままでは直流成分に不要なバイアスによりダイナミックレンジが低下してしまうので、最小値の減算という処理を行う。そして、最小値減算後の最小値と最大値との間のピクセル値をそのまま使用するようにする。このようにして、ステップS 1 5の撮影処理が実行され、内部RAM(図示せず)のテンポラリデータ配列にはコントラストを高めた撮影データ(ピクセル値)が格納される。

そして、図1 3に戻って、この図1 3のステップS 1 7において、ゲームプロセサ6 4は、そのピクセル値に基づいて、撮影部カバー2 6の裏面の4つの黒マーク6 2を検出する。そして、ステップS 1 9で、図2 1に示すように4つのマークをすべて検出したかどうか判断する。このステップS 1 9で“YES”が判断されると、そのときにはカード挿入口2 8(図1および図3)にカードが挿入されておらず、したがって、カバー2 6の裏面がそのまま撮影されたことを意味する。そして、この場合には、図2 2に示すように、黒色マーク6 2の位置のピクセル9 4が最も黒くなる。プロセサ6 4は、続くステップS 2 1において、図2 2に示す4つのピクセル9 4で決まる矩形領域の中心9 6の座標を求める。そして、次のステップS 2 3では、上記中心座標9 6を中心として、 24×24 の正方形領域を撮影画素領域すなわち有効領域9 8として設定する。

このようにして、システム初期化処理において、 32×32 画素から 24×24 画素の有効画素領域9 8を設定できるので、たとえば組み立て誤差などによってイメージセンサ5 4や他の部分に位置ずれが仮に生じていたとしても、そのような位置ずれが補正または校正され、常に、正しい位置からのピクセル値が取り込めることになる。

なお、この実施例では回転方向についての位置ずれは補正していないが、必要に応じて角度補正することも可能である。しかしながら、この実施例では、後に説明するように、元の $576(24 \times 24)$ ピクセルを6 4個のリサンプリングピクセル(3×3)に解像度を粗くしてリサンプリングすることによって、多少の角度ずれは補償することができる。

また、実施例では4つのマークを使って中心位置およびそれを含む所定ピクセルの有効画素領域を設定するようにした。しかしながら、マークは最低対角の2つあれば同様に中心および有効領域を求めることができる。

図1 2のステップS 1の後、ゲームプロセサ6 4は、ステップS 2でゲームモード選択画面を図1に示すテレビジョンモニタ1 4上に表示させる。そして、次のステップS 5でゲームモードが決定されるのを待つ。そして、決定されたゲームモードでのゲームプレイが実行され(ステップS 7)、次のステップS 9でその結果が表示される。

図1 2に戻って、図1 2のステップS 7で示すゲームプレイルーチンが図2 3に詳細

に示され、その最初のステップS113で、ゲームプロセサ64は、ゲーム処理を開始する。そして、次のステップS115で、テレビジョンモニタ14上に、たとえば図31に示すように「カードを撮影部にセットして撮影キーを押してください」のメッセージを表示して、ゲームプレイヤにカードをカード挿入口28（図1）に挿入するように促す。そして、ステップS117で、決定および撮影キー42が押されたことを検出すると、次のステップS119で、カード撮影および識別ルーチンを実行する。

カード撮影および識別ルーチンは詳しくは図24に示され、その最初のステップS139では、ゲームプロセサ64は、先に説明した図18に示す撮影ルーチンを実行する。したがって、ステップS139が実行されたときに、テンポラリデータ配列の各要素（576ピクセル）に、それぞれのピクセル値が格納されている状態である。

このような状態で、ゲームプロセサ64は、次のステップS141で、撮影結果が「カードなし」の状態に一致するかどうか判断す。そして、ステップS143で「カードなし」つまり“YES”を判断した場合には、続くステップS145で、ゲームプロセサ64は、返り値（サブルーチンからサブルーチンを呼び出した処理へ返す結果のこと）を「カードが置かれていない。」として設定する。

ステップS143で“NO”を判断したときには、次のステップS147で、リサンプリングを実行する。リサンプリングをする理由は、必要な識別精度を確保しながら、他方で識別のためのプロセサ64の処理負荷やデータベース67Aのサイズを最小化することである。このリサンプリングルーチンは詳細には図25に示される。先に概説したとおり、リサンプリングにおいては、図26に示すように、元の576（ 24×24 ）のピクセル、すなわち $P[0][0] - P[23][23]$ を64個のリサンプリングピクセル $Q[0] - Q[63]$ に変換する。その場合、図26に示すような2次元窓関数を使う。そのような2次元窓関数を使うことによって、折り返しノイズが除去できるばかりでなく、カードの位置ずれを吸収しまたは補償することができる。

実施例の2次元窓関数では、 3×3 の9つの元のピクセルの内、その中心ピクセルのピクセル値を「4」の重みで取り扱い、それに近い4つのピクセルを「2」の重みで取り扱い、中心ピクセルから最もはなれた4つの角のピクセルを「1」の重みで扱う。これは、リサンプリングピクセルを形成する元のピクセルのなかで中心に近いピクセルほど重み付けを重くすることによって、中心に近いピクセルを重視するためである。

このようなリサンプリングのために、図25の最初のステップS159では、インデックスカウンタI（図示しないが内部RAMに形成されている）に初期値「0」を設定する。

その後ステップS161において、 $Q[I]$ に対応する9つのピクセルをロードし、それぞれ $P0 - P8$ のローカル変数に格納する。

ここで、 $P0$ は図26で示す9個のピクセルの左上端のピクセルを指し、以下順に、右へ $P1$ 、 $P2$ と進み、左中が $P3$ となり、中心が $P4$ 、その右が $P5$ となり、左下から右へ順に $P6$ 、 $P7$ 、 $P8$ となるように割り付けられている。なお、ステップS161で整数除算やモジュロ演算を使用しているのは、テンポラリデータ配列が2次元配列で各ピクセル値を記憶している一方、リサンプリングピクセル $Q[0] - Q[63]$ は1次元レジスタにロードする必要があるからである。このようにして、このステップS161では、メモリから各元のピクセル値を読み出す。

続くステップS163で、プロセサ64は、先に説明した2次元窓関数の重み付けに

従って9個の元のピクセル値の総和を計算する。つまり、ピクセルP 0, P 2, P 6, P 8に「1」を掛け算し、P 1, P 3, P 5, P 7に「2」を掛け、P 4に「4」を掛けた後に、それらの総合計Qを計算する。そして、ステップS 1 6 5では、この総和Qをリサンプリングピクセル配列（図2 6）のQ [I] の判定用画素データとして格納する。次のステップS 1 6 7でインデックスIをインクリメントし、ステップS 1 6 9において“YES”すなわちインデックスIが「6 3」になったことを検出するまで、ステップS 1 6 1－S 1 6 5が繰り返し実行される。したがって、6 4個のすべてのリサンプリングピクセルQ [0]－Q [6 3]についての判定用画素データすなわちリサンプリングピクセル値がメモリ内に格納される。

このようにしてステップS 1 4 7（図2 4）のリサンプリングプロセスを実行した後、次のステップS 1 4 9では、図2 7のサブルーチンに従って、直流成分除去および正規化処理を実行する。この直流成分除去および正規化処理は、カード毎に図柄全体の明度レベルやコントラストが異なるため、識別精度を高める目的で行われ、識別用画素データ配列（リサンプリングピクセル値）の直流成分を除去し正規化する。

すなわち、図2 7のステップS 1 7 1では、全ピクセルデータ（ここでは、リサンプリングピクセル値のこと）の平均値を算出し、ステップS 1 7 3で直流成分の除去を行う。つまり、各リサンプリングピクセル値を「リサンプリングピクセル値－平均値」として計算し直す。次いで、ステップS 1 7 5で、各リサンプリングピクセル値の絶対値の平均を計算することによって、平均偏差を求める。その後、ステップS 1 7 7で飽和演算（{リサンプリングピクセル値×正の最大値／2} ÷ 平均偏差）を実行することによって、直流成分除去後の各リサンプリングピクセル値を正規化する。

このようにして、図2 4のステップS 1 4 9が終わった段階で、内部RAMには、図柄判定のための6 4個のリサンプリングピクセル値が格納されている。その状態で、続くステップS 1 5 1で、図7に示すデータベース6 7 A（および図8に示すデータベース6 7 B）を参照または検索して、そのとき撮影したカードがどのカードであるかを特定する。このステップS 1 5 1では、識別用画素データ配列（リサンプリングピクセル値）と判定用画素データ配列（データベース6 7 A）との間の類似性を判断するために、たとえばユークリッド距離を用い、その距離の最も小さい判定用（比較用）画素データ配列のカードIDを、そのとき撮影したカードのカードIDとして特定する。なお、データベース検索のサブルーチンは具体的には図2 8および図2 9に示される。

図2 8の最初のステップS 1 7 9では、インデックスカウンタxおよびyに「0」を設定する。インデックスxは図7に示すデータベース6 7 Aのエントリ番号を示すインデックスであり、インデックスyは識別候補数を示すインデックスである。

次のステップS 1 8 1では、プロセサ6 4は、判定用画素データ配列Qの各要素（リサンプリングピクセル値）と図7のデータベース6 7 Aのx番目のエントリA xの各要素との差の2乗を求めて総和を計算する。つまり、この実施例では、一致検出のアルゴリズムとして、識別用画素データ配列の各要素すなわちリサンプリングピクセル値とエントリの各要素との差をデータベースで予め設定しているすべてのエントリ（実施例ではエントリ# 0－# 1 2 7：図7）についてそれぞれ算出し、その総和が最も小さいエントリを、一致したカードのID（識別番号）として特定する方式を採用する。ただし、各要素ごとの差を強調して識別を容易にするために、実施例では差の2乗の総和Dを計算するようにしている。つまり、差の絶対値の総和を求めてもよく、差の2乗の総和を

求めてもよい。

また、このステップS181で求める総和Dは、多次元データ配列の類似性を判断するための「距離」の概念に含まれるもので、具体的には、ユークリッド距離に相当する。そして、このような距離としては、ユークリッド距離の他に、ハミング距離等が利用されてもよい。

続くステップS183では、プロセサ64は、ステップS181で求めた総和Dが所定の閾値より小さいかどうか判断する。これは、この総和Dが一定値以上の場合にはデータベース内のリファレンス値との差が大きすぎるので、候補として残さないようにするためである。

ステップS183で“YES”が判断されると、総和Dが閾値より小さいので、上記xをカードIDとして総和Dの値とともに、構造体配列R[y]の要素としてメモリ内に格納する。そして、ステップS187で候補数yをインクリメントするとともに、ステップS189でデータベース67Aのエントリを進めるべくエントリ番号xをインクリメントする。そして、ステップS191で、エントリ番号xがデータベース67Aの総エントリ数（実施例の場合には128）に達したかどうか判断する。“NO”なら、ステップS181-S189を繰り返し実行する。そして、判定用画素データ配列のすべての要素（リサンプリングピクセル値）の処理を終了すると、ステップS191で“YES”が判断され、続くステップS193に進む。

ステップS193では、プロセサ64は、識別候補数yが「0」かどうか判断する。このステップS193で“YES”が判断されるということは、全てのエントリに対して総和Dが閾値を超えてしまう場合であり、この場合には、次のステップS195で「一致カードなし」の返り値を設定し、図24のステップS153にリターンする。

ステップS193で“NO”が得られると、次のステップS197で、候補数yが「1」かどうか判断する。候補数が1つであるということは、その候補が特定すべきカードに他ならない。したがって、その場合には、次のステップS199において、その候補構造体R[0]のカードIDを返り値として設定して、ステップS153（図24）に戻る。

先のステップS197で“NO”が判断されるということは、識別候補数yが「2」以上であること、つまり、複数の識別候補が残ってしまった場合である。このような場合には、どれか1つの候補を絞る必要があるので、それ以後は、図29のフロー図の処理に移る。

図29のステップS201では、プロセサ64は、y個の識別候補RをステップS181（図28）で計算した総和Dの小さい順にソーティングし、次のステップS203でインデックスカウンタyに「1」を設定する。つまり、Dの小さい順に並べた識別候補Rの中で、第2番目に小さいDの候補を指定する。最小のDの候補がR[0]である。

そして、ステップS205において、プロセサ64は、そのときの候補R[1]と最小Dの候補R[0]とのDの差DifAを計算する。つまり、このステップS205では、各候補R[y]と最小候補R[0]との間のDの差DifAを計算する。そして、ステップS207において、その差DifAが所定の閾値Aより大きいかどうか判断する。つまり、Dの最小の候補構造体R[0]とそのときの候補構造体R[y]との似ている程度が閾値Aより大きいかどうか、つまり候補構造体R[y]を候補として残す意味があるかどうか判断する。

このステップS 2 0 7で“NO”が判断されると、つまり、候補として残すべきであると判断したときには、続くステップS 2 0 9で、今度は、ステップS 2 0 1でソートしたDの小さい順に並んだ各候補構造体の隣接するものどうしの間のDの差D i f Bを計算する。つまり、候補R [y] のDからそれぞれより1つ前の候補R [y - 1] のDを減算する。そして、ステップS 2 1 1において、その差D i f Bが所定の閾値Bより大きいかどうか判断する。つまり、Dの小さい順に並べた候補の隣接する候補どうしの似ている程度が閾値Bより大きいかどうか、つまり候補構造体R [y] を候補として残す意味があるかどうか判断する。

ステップS 2 0 7またはステップS 2 1 1で“YES”が判断されたときには、上述のように、その候補構造体R [y] を候補として残す意味がないので、次のステップS 2 1 3で、R [y] 以降の構造体を識別候補から除外する。

そして、ステップS 2 1 5で、その結果候補構造体Rが1つだけ残ったかどうか判断する。このステップS 2 1 5で“NO”が判断されるということは複数の候補構造体が残ったということであり、その場合には、ステップS 2 1 7で、残った候補のカードIDの組み合わせに一致するエントリが存在するかどうか、図8のデータベース6 7 Bを参照して検索する。つまり、ステップS 2 1 7では、残った2以上の構造体Rがデータベース6 7 Bのどれかと一致するかどうか判断する。データベース6 7 Bには、先に説明したように、識別候補として残ったカードIDの組み合わせを順番にリストアップしている。そして、ステップS 2 1 7では、残った複数の候補のカードIDの組み合わせが、データベース6 7 BにリストされているカードIDの組み合わせパターンと一致するものがあるかどうかを検索する。そして、ステップS 2 1 9で、パターンが一致するエントリがあるかどうか判断する。

ステップS 2 1 9での判断が“YES”の場合には、その一致したパターンを有するエントリが示すカードの識別番号 (ID) を返り値として設定し、図2 4のステップS 1 5 3にリターンする。

ステップS 2 1 9で“NO”が判断されたときには、先のステップS 2 1 5で“NO”が判断されたときと同様に、ステップS 2 2 3において、ステップS 2 0 1でソーティングした最上の、Dが最も小さい候補R [0] のカードIDを返り値として設定して図2 4のステップS 1 5 3にリターンする。

なお、ステップS 2 1 1で“NO”が判断された場合、すなわち、各構造体の最小Dの構造体R [0] との差D i f Aが閾値Aより小さく、しかもソーティングしたときの隣接構造体どうしの差D i f Bが閾値Bより小さいときには、ステップS 2 2 5において、次の (第3番目以降の) 構造体を指定するようにインデックスyをインクリメントする。そして、次のステップS 2 2 7ですべ手の候補構造体について処理したかどうか判断する。したがって、ステップS 2 2 7で“NO”ならステップS 2 0 5に戻ってステップS 2 0 5 - S 2 1 1を繰り返し実行する。ステップS 2 2 7で“YES”なら、ステップS 2 1 7に進む。

このようにして、図2 4のステップS 1 5 1においてデータベース6 7 A (およびデータベース6 7 B) を検索することによって、ステップS 1 9 5、S 1 9 9 (ともに図2 8)、ステップS 2 2 1、S 2 2 3 (ともに図2 9) のいずれかの返り値が検索結果として戻される。そして、ステップS 1 5 3において、ゲームプロセサ6 4は、返り値が「一致カードなし」かどうか、すなわちステップS 1 9 5から返り値があったかどうか判断

する。“NO”の場合にはステップS155において、戻り値として、ステップS199、S221またはS223で特定したエントリのカードIDを戻り値として設定し、ゲームプレイルーチン(図23のステップS121)にリターンする。“YES”の場合には、「一致カードなし」の戻り値を設定してステップS121にリターンする。

図23に戻って、ステップS121において、先のステップS119からの戻り値が「カードが置かれていない」かどうか判断する。“YES”の場合には、先のステップS115に戻る。“NO”の場合には、次のステップS123に進み、戻り値が「一致カードなし」かどうか判断する。このステップS123で“YES”が判断されたときには、次のステップS137において、メッセージ「識別できないカードです。正しいカードを撮影部に入れて撮影キーを押してください」をテレビジョンモニタ14上に表示し、ゲームプレイヤにカードの交換を要請する。

そして、ステップS123で“NO”が判断されると、次のステップS125で、プロセッサ64はそれぞれの場合の戻り値に示されるカードIDを受け取る。その後、ステップS127で、プロセッサ64はデータベース67Aから、そのカードIDで示されるカードの表示画面100を図30に示すようにテレビジョンモニタ14上に表示する。ここで、表示画面100は、図柄部102を含み、この図柄部102には、カード30(図2)の図柄31と同様の図柄が表示される。これによって、ゲームプレイヤは、自分がそのときカード挿入口28へ挿入したカードが正しく認識されたことを知ることができる。カード画面100はさらに、名称部104を含み、この名称部104には、図柄部102に表示された図柄を表現する名称、この実施例では「ドラゴン」が表示される。カード画面100に含まれる説明文106は、その図柄で表現されるかつその名称で呼ばれるもの(キャラクタ)がどのような特性のものであるかを説明する文章で表され、さらにパワー表示部108には、その名称で示されるものの攻撃および防御のパワー、実施例の場合には攻撃力530000が表示される。

このようにして、識別したカードに対応するカード画面100が表示された後、ステップS129で、ゲームプロセッサ64は、コンピュータプレイヤとして行動し、そのコンピュータプレイヤが対戦のために出すカードのカードIDを決定し、ステップS131で、両方のプレイヤ、すなわち実際のゲームプレイヤとコンピュータプレイヤが出したカードによって、そのターンの勝敗を決定し、ステップS133でターンの終了を検出すると、ステップS135において、ゲーム画面110を図31に示すように、テレビジョンモニタ上に表示する。このゲーム画面110では、ゲームプロセッサが演じるコンピュータプレイヤを示すプレイヤキャラクタ112とともに、そのコンピュータプレイヤが出したカードを示すカードキャラクタ114が表示される。ただし、ターン表示部116は現在のターンを表示し、メッセージ表示部118には、たとえばゲームプレイヤにカードの挿入を促すメッセージ「カードを撮影部にセットして撮影キーを押してください」が表示される。

なお、当然のことであるが、カードゲームそれ自体(たとえばゲームルールや具体的なカードの図柄など)は適宜変更され得るものであり、それに応じて、ゲーム画面110やカード画面100は適宜変更され得る。

さらに、上述の実施例はカードを識別し、その識別結果に応じた入力を得て対戦ゲームをプレイするカードゲームシステムを例に挙げて説明した。しかしながら、この発明に従ったカード識別は、そのようなカードゲーム以外の他の娯楽(エンターテインメン

ト) システム、たとえばカラオケなどにも適用可能であることはいうまでもない。

さらに、上述の実施例は、カードの図柄を撮影しリサンプリング等の処理を施した画素データ配列に基づき、データベースに格納された比較用データ配列を検索して一致または最も近いエントリを検索して、カードIDを出力として得るものである。しかしながら、撮影して得られた画素データ配列またはリサンプリング後のデータ配列に対して一般によく知られる文字認識等の処理を施して文字列データを得ることも可能である。

また、上述の実施例のカード判別処理と文字認識処理とを組み合わせ、カードIDと文字列データとを同時に得るようにすることも可能である。

この発明が詳細に説明され図示されたが、それは単なる図解および一例として用いたものであり、限定であると解されるべきではないことは明らかであり、この発明の精神および範囲は添付されたクレームの文言によってのみ限定される。

請求の範囲

1. それぞれに人間が視覚的に識別可能な図柄が印刷されている複数のカードから入力を得て、その入力に応じた情報処理を行うエンターテインメント装置であって、

前記カードの前記図柄を撮影して撮影画素データ配列を得る撮影手段、

前記複数のカードの各々に個別的に対応する複数のエントリを含み、各々のエントリは1組のカードIDおよび比較用データ配列を含むデータベース、

前記データベースから前記撮影画素データ配列に基づいて特定の比較用データ配列を検索し、その特定の比較用データ配列と組になっているカードIDを取得するカード識別手段、および

前記カード識別手段によって取得された前記カードIDを入力として前記情報処理を実行する情報処理手段を備える、カードを用いるエンターテインメント装置。

2. 前記撮影手段は前記図柄を撮影して撮影信号を出力するイメージセンサ、前記撮影信号を第1解像度でサンプリングしてデータ配列を形成するデータ配列形成手段、および前記データ配列を前記第1解像度より低い第2解像度でリサンプリングして前記撮影画素データ配列を形成する撮影画素データ配列形成手段を含み、

前記比較用データ配列は前記第2解像度に対応する比較用データを含む、請求項1記載のエンターテインメント装置。

3. 前記カード識別手段は、前記撮影画素データ配列および前記比較用データ配列の距離を算出し、その距離の最も小さい比較用データ配列のエントリのカードIDを取得する、請求項2記載のエンターテインメント装置。

4. 前記距離は、前記撮影画素データ配列の各要素と、前記各要素に対応する前記比較用データ配列の各要素との差分の絶対値の総和である、請求項3記載のエンターテインメント装置。

5. 前記距離は、前記撮影画素データ配列の各要素と、前記各要素に対応する前記比較用データ配列の各要素との差分の2乗の総和である、請求項3記載のエンターテインメント装置。

6. 前記撮影画素データ配列形成手段は、前記データ配列の各要素に対して所定の重み付けを行って前記撮影画素データ配列を形成する、請求項2ないし5のいずれかに記載のエンターテインメント装置。

7. 前記カード識別手段は、前記差分総和が所定の閾値より大きいかどうか判断する閾値判断手段を含み、前記差分総和が前記所定の閾値より大きいエントリは識別候補から排除する、請求項2ないし6のいずれかに記載のエンターテインメント装置。

8. 前記カード識別手段は、前記閾値判断手段の判断の結果残った候補の総数を判断する候補数判断手段を含み、前記候補数判断手段が候補数「0」を判断したとき、カードIDを取得せず、候補数「1」を判断したとき、その識別候補のカードIDを取得する、請求項7記載のエンターテインメント装置。

9. 前記データベースを第1データベースとし、

1以上のエントリを含み、各々のエントリは複数の候補カードIDおよび1つの判定カードIDを含む第2データベースをさらに備え、

前記カード識別手段は、前記識別候補が2以上残ったかどうか判断する候補数判断手段を含み、さらに前記候補数判断手段が候補数「2以上」を判断したとき、前記残った候補のカードIDの組み合わせと前記候補カードIDの組み合わせが一致する前記第2

データベースのエントリを検索し、一致するエントリが存在する場合に当該エントリの判定カードIDを取得する、請求項8記載のエンターテインメント装置。

10. 前記データベースは各エントリ毎にそれに対応するカードデータを含み、さらに

前記情報処理手段は前記カード識別手段によって取得された前記カードIDに対応するエントリのカードデータに基づいて少なくとも図柄を表示するカードデータ表示手段を含む、請求項1ないし9のいずれかに記載のエンターテインメント装置。

11. カートリッジコネクタをさらに備え、前記カートリッジコネクタにはメモリカートリッジが装着され、そのメモリカートリッジには別のデータベースが格納される、請求項1ないし10のいずれかに記載のエンターテインメント装置。

12. それぞれに人間が視覚的に識別可能な図柄が印刷されている複数のカードから入力を得て、その入力に応じた情報処理を行うエンターテインメント装置であって、

前記カードの前記図柄を撮影して撮影画素データ配列を得る撮影手段、

前記撮影画素データ配列から図柄に対応するデータ列を取得するカード識別手段、および

前記カード識別手段によって取得された前記データ列を入力として前記情報処理を実行する情報処理手段を備える、カードを用いるエンターテインメント装置。

13. 前記カードを所定位置にセットするカード撮影部、および

前記カード撮影部にセットされたカードの被撮影面に光を照射するための光源をさらに備える、請求項1ないし12のいずれかに記載のエンターテインメント装置。

14. 前記光源からの光を拡散反射させて前記被撮影面に入射させる反射手段をさらに備える、請求項13記載のエンターテインメント装置。

15. 前記カード撮影部を覆い、前記撮影手段と対向する面に位置補正用マークを備える撮影部カバー、および

前記位置補正用マークに基づいて撮影画素データの取得領域を補正する撮影画素データ取得領域補正手段をさらに備え、

前記撮像手段は、前記カード撮影部にカードがセットされていない状態では、前記位置補正用マークを撮影する、請求項13または14記載のエンターテインメント装置。

16. それぞれに人間が視覚的に識別可能な図柄が印刷されている複数のカードを撮影することによってカードを特定する方法であって、

(a) 前記複数のカードの各々に個別的に対応する複数のエントリを含み、各々のエントリは1組のカードIDおよび比較用データ配列を含むデータベースを準備し、

(b) イメージセンサによって前記図柄を撮影して撮影信号を得て、

(c) 前記撮影信号を第1解像度でサンプリングしてデータ配列を形成し、

(d) 前記データ配列を前記第1解像度より低い第2解像度でリサンプリングして撮影画素データ配列を形成し、そして

(e) 前記データベースから前記撮影画素データ配列に基づいて特定の比較用データ配列を検索し、その特定の比較用データ配列と組になっているカードIDを取得する、カード特定方法。

17. 前記ステップ(e)では、前記撮影画素データ配列および前記比較用データ配列の距離を算出し、その距離の最も小さい比較用データ配列のエントリのカードIDを取得する、請求項16記載のカード特定方法。

図2

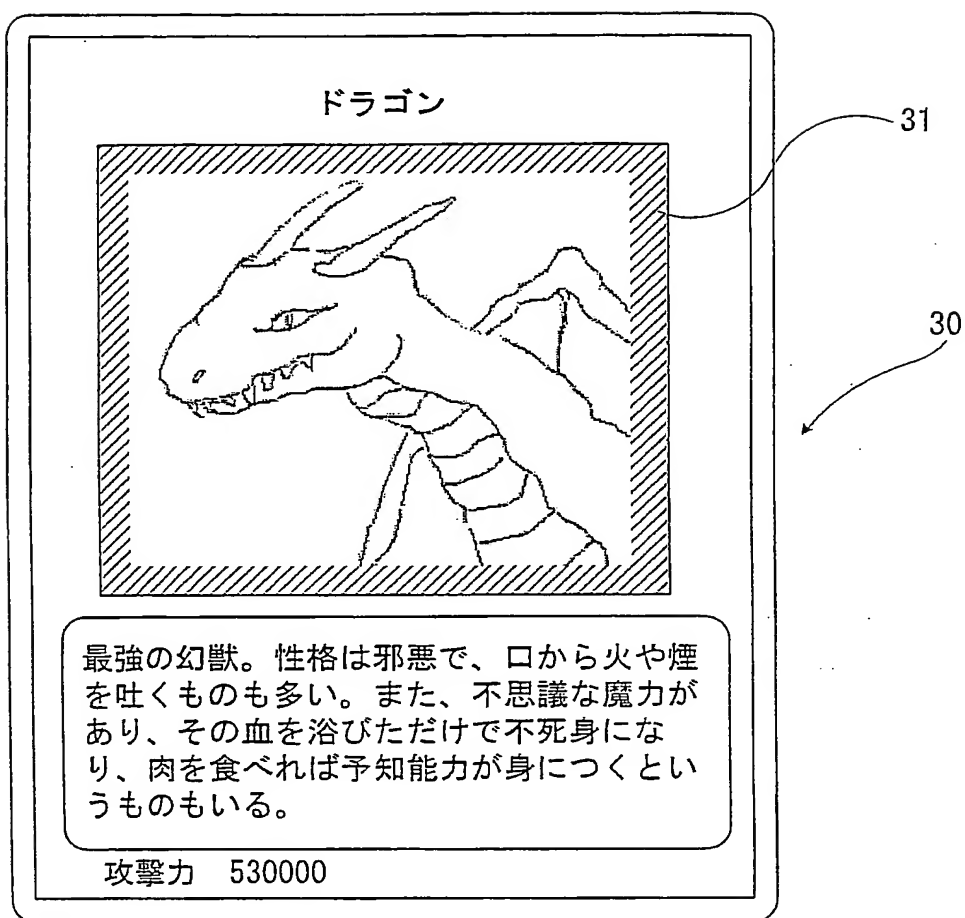


図 3

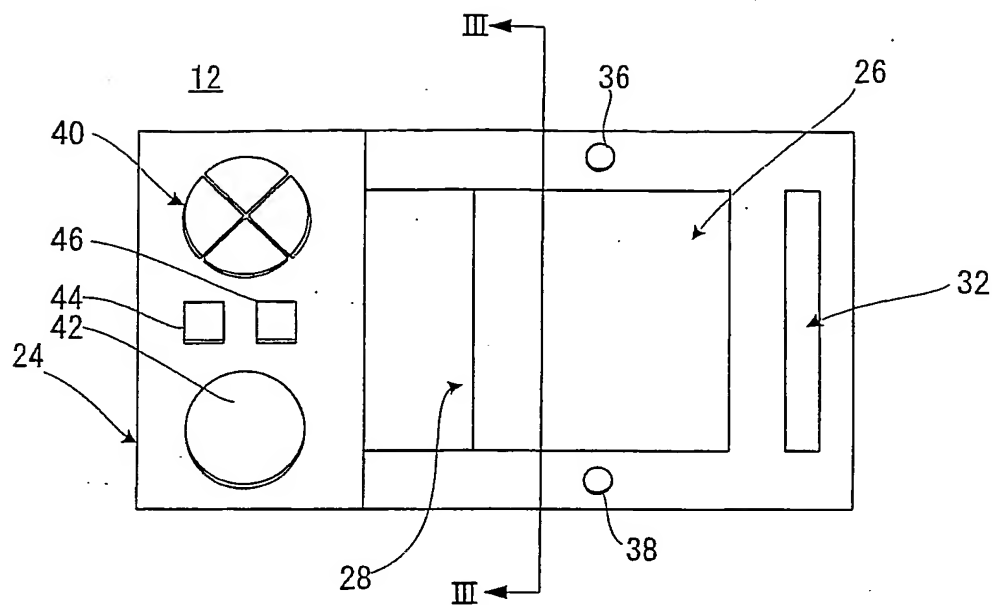


図 4

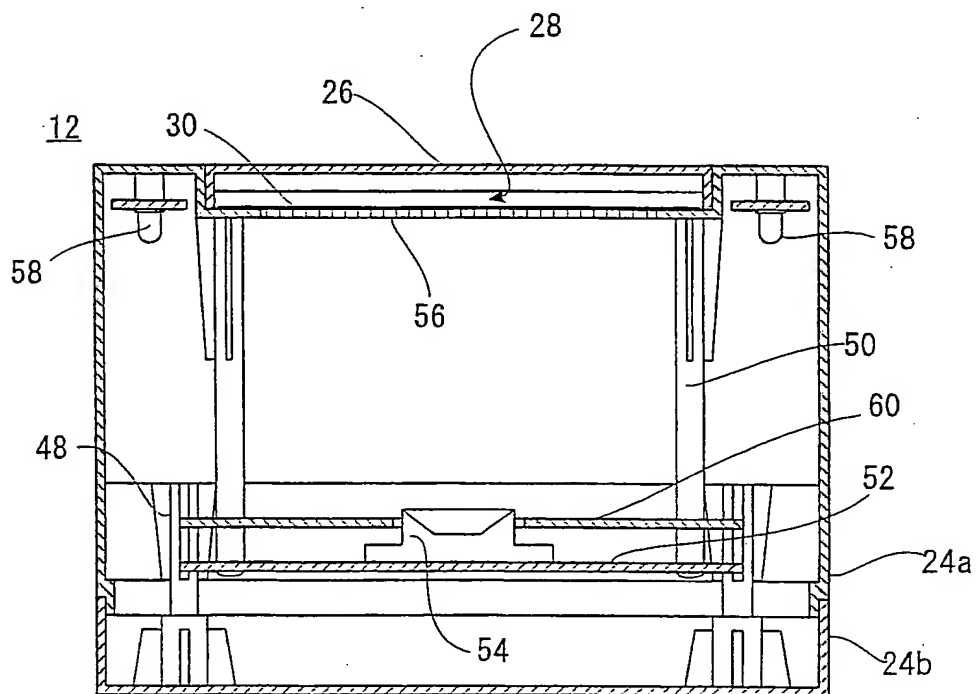
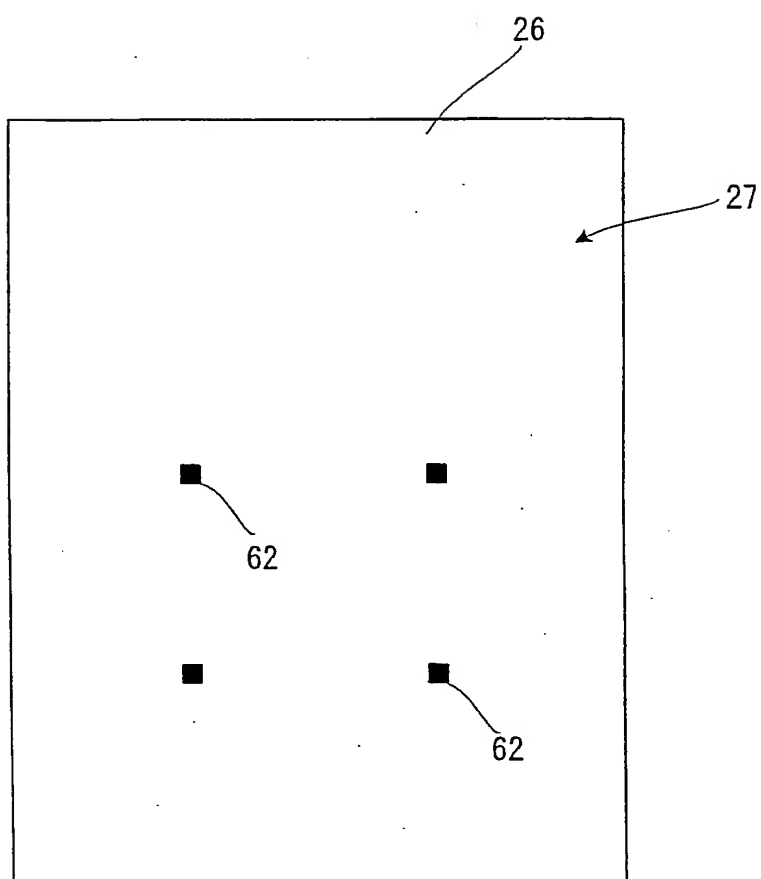


図 5



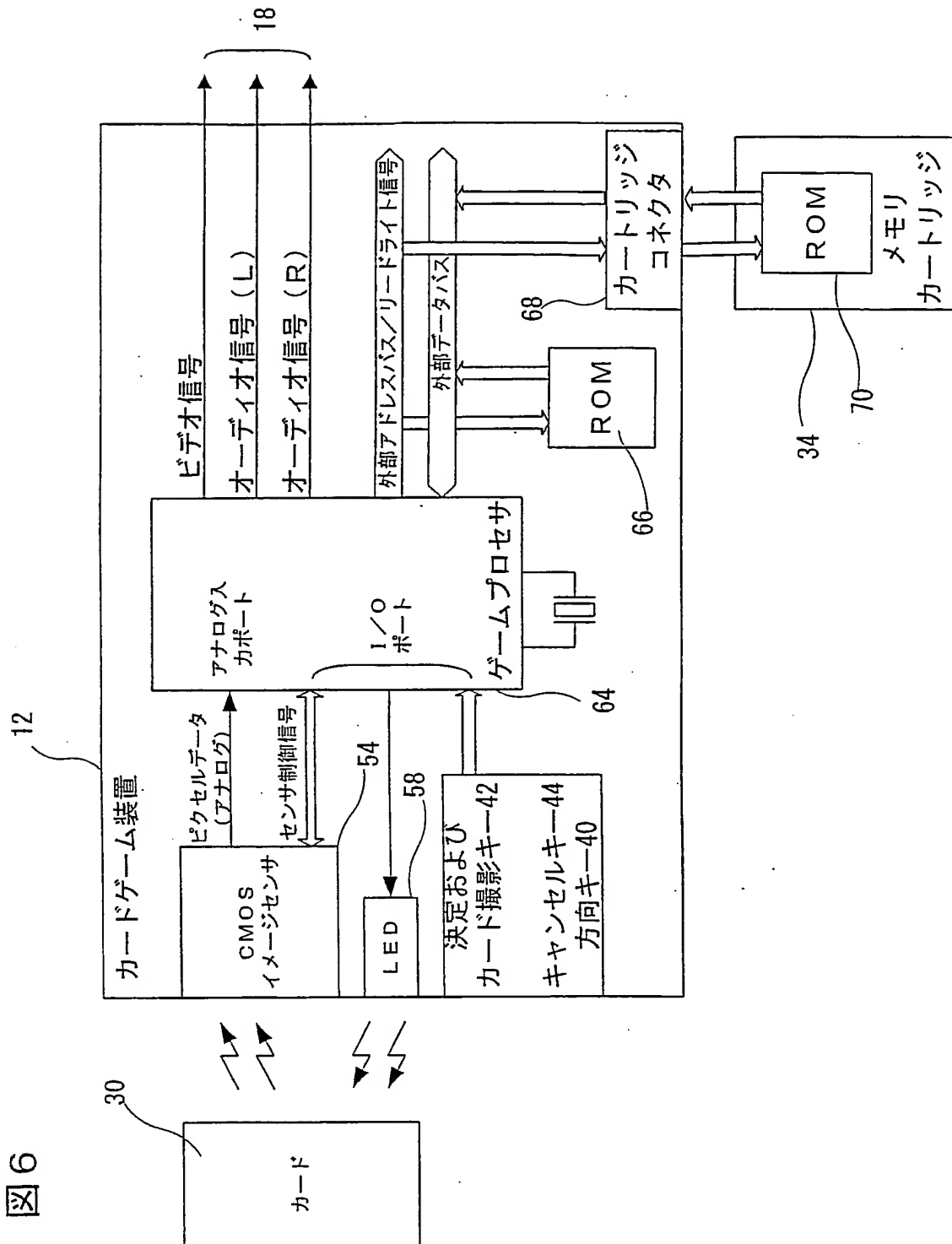


図 7

67A

	[0]	[1]	[2]	.	.	.	[63]	
エントリ#0	06	27	A8	.	.	.	6D	カードデータ
エントリ#1	00	36	98	.	.	.	43	カードデータ
エントリ#2	25	66	74	.	.	.	AC	カードデータ
.	
.	
.	
エントリ#127	B8	F1	58	.	.	.	36	カードデータ

図 8

67B

	識別 カードID	ID[0]	ID[1]	ID[2]	ID[3]	.	.	.
エントリ#0	#1	#1	#12	#18				
エントリ#1	#3	#3	#1	#110	#65			
エントリ#2	#4	#22	#4					
.	.	.	.					
.	.	.	.					
.	.	.	.					
エントリ#85	#122	#122	#127					

図 9

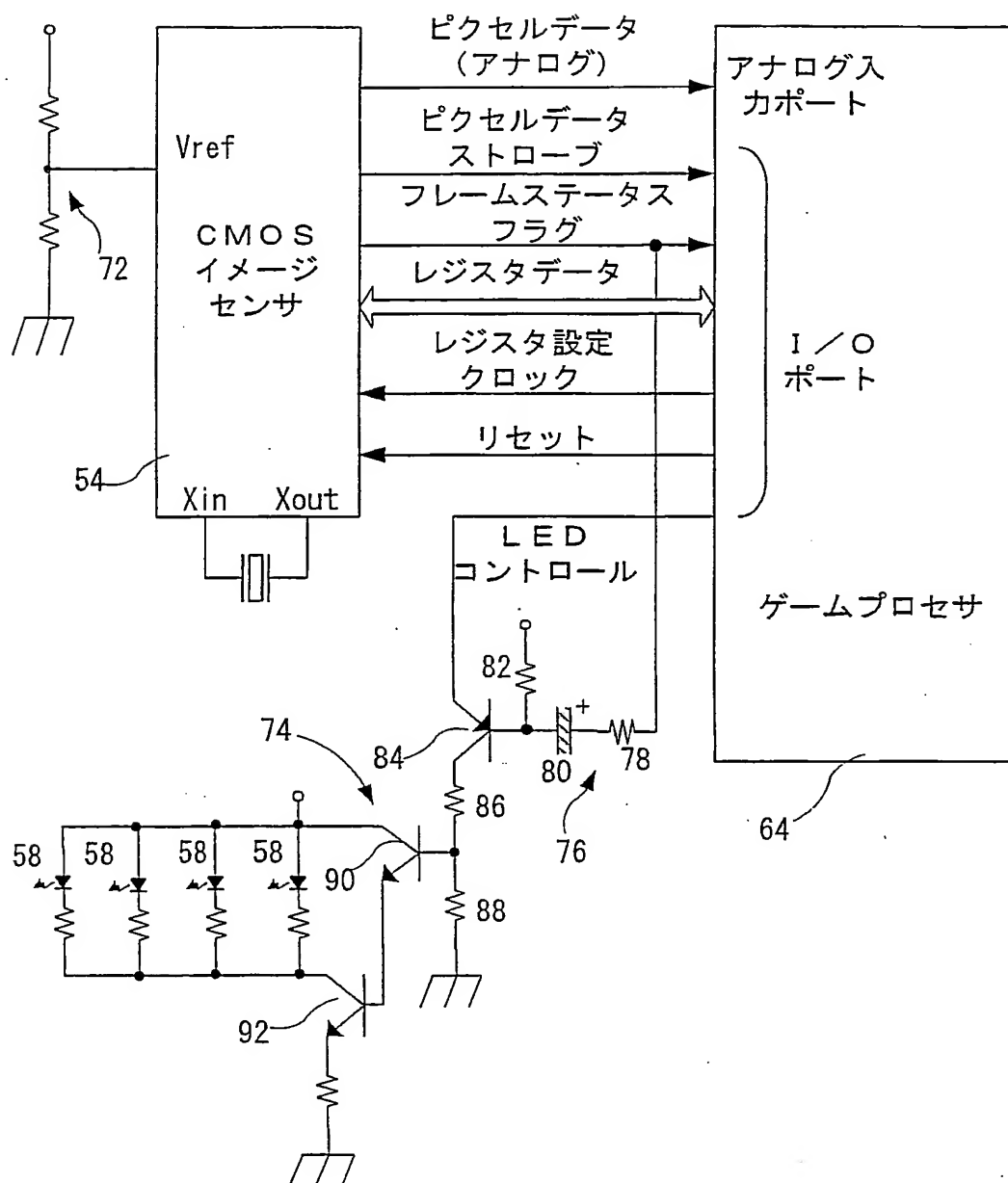


図 10

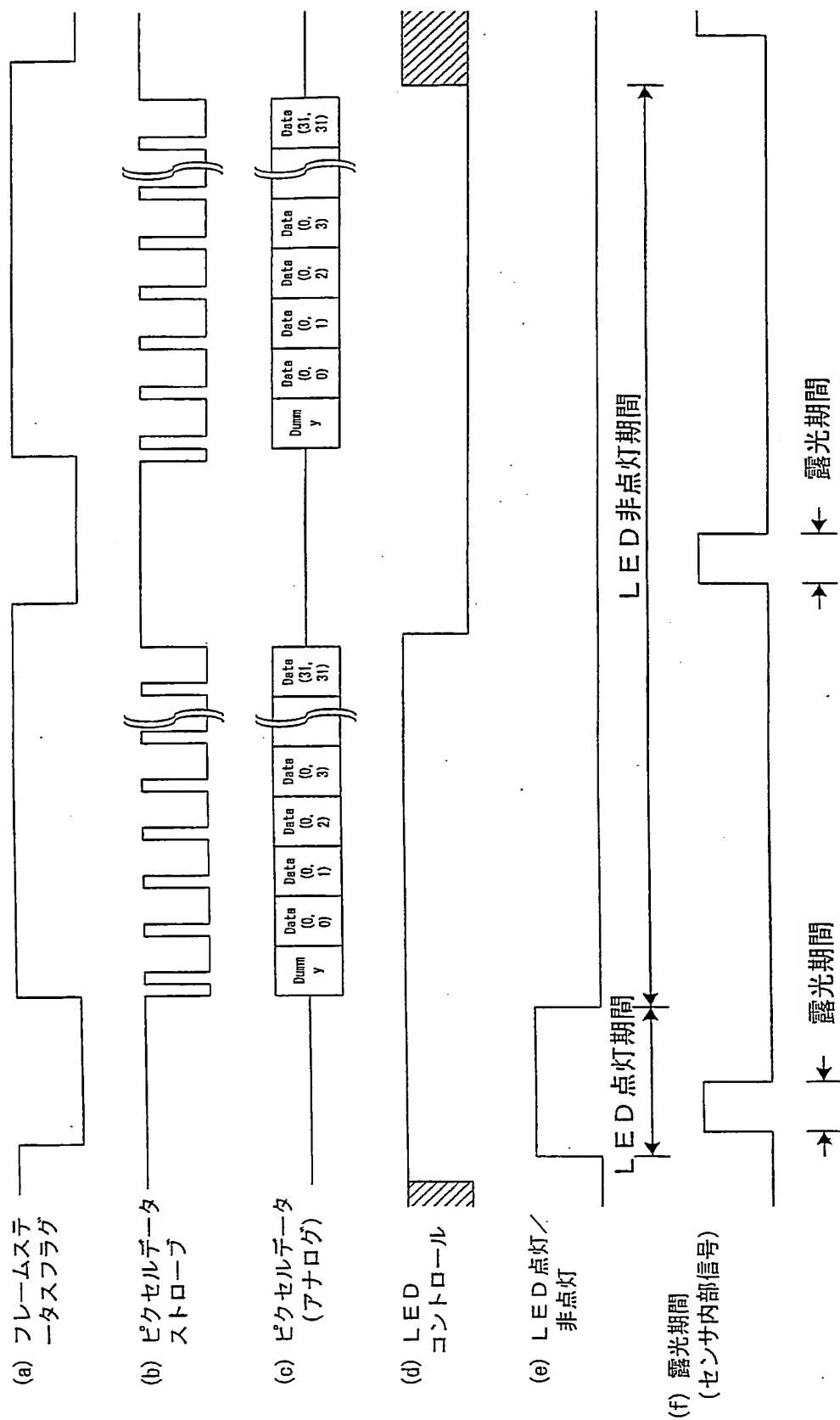


図 11

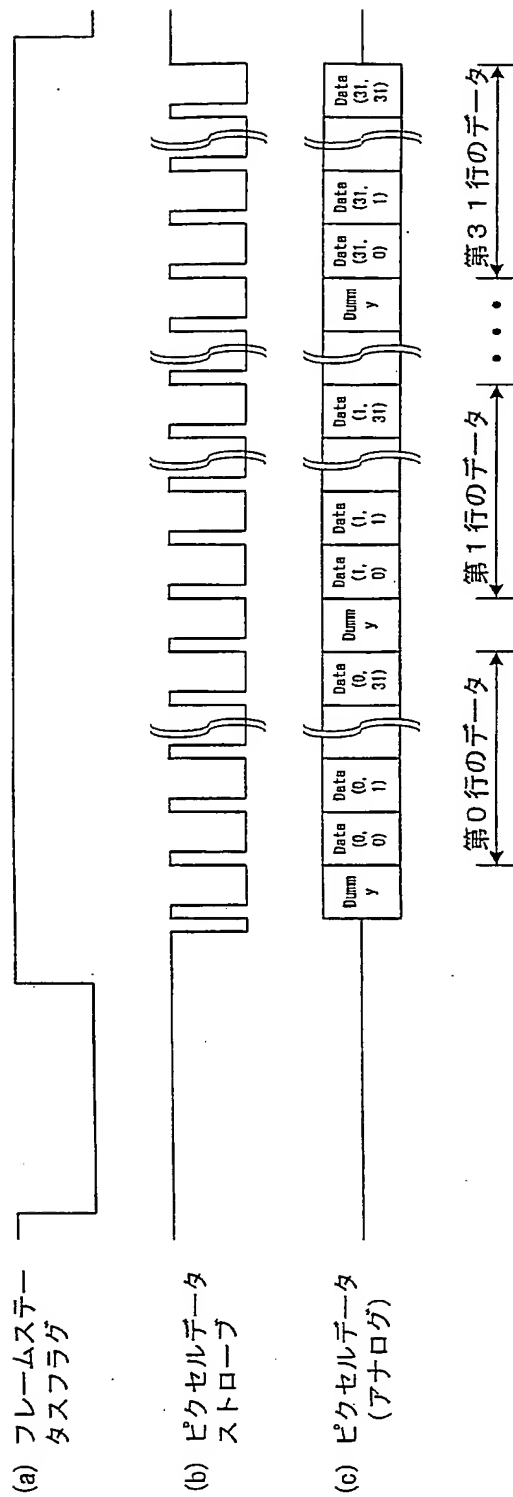


図 1 2

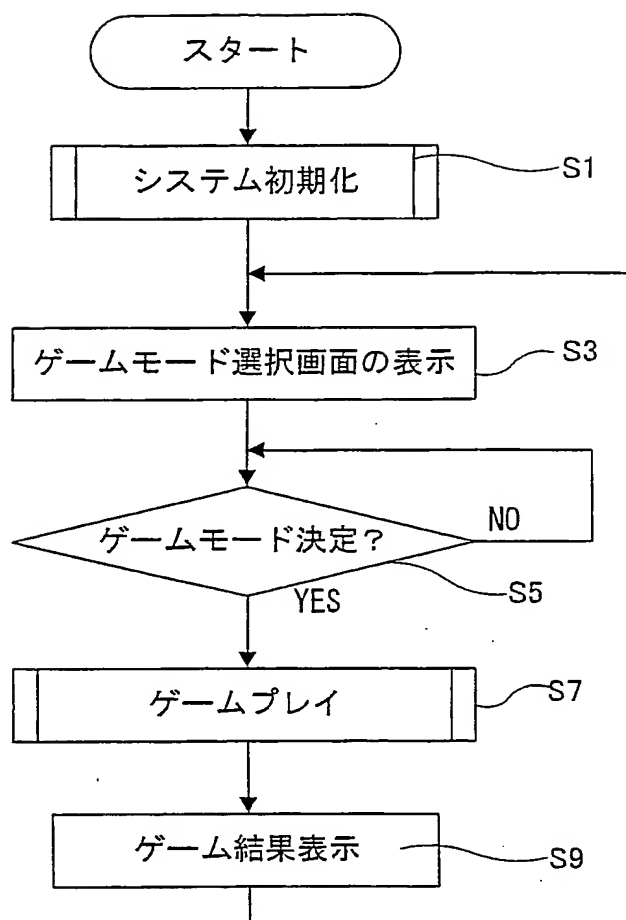


図 13

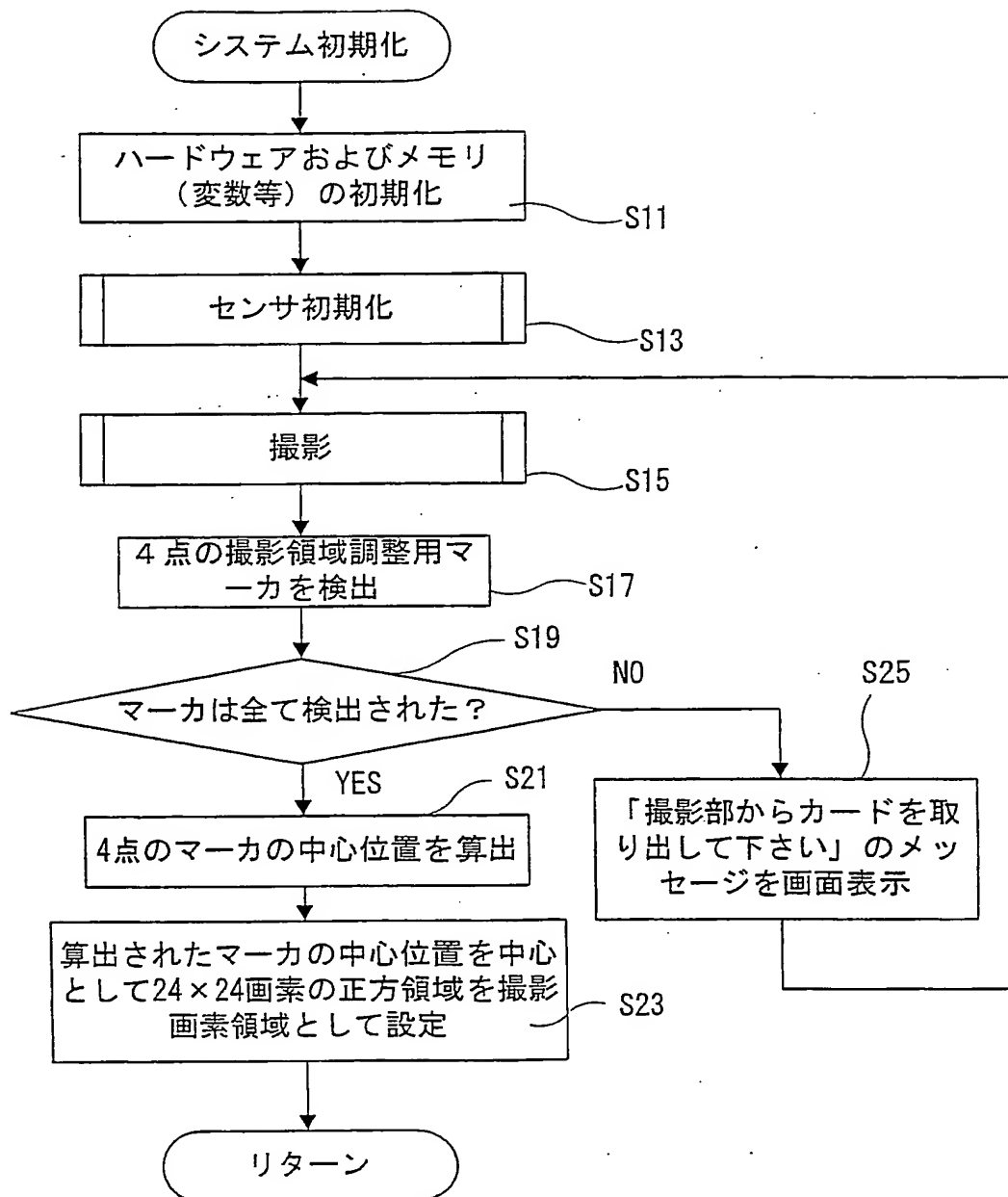


図 1 4

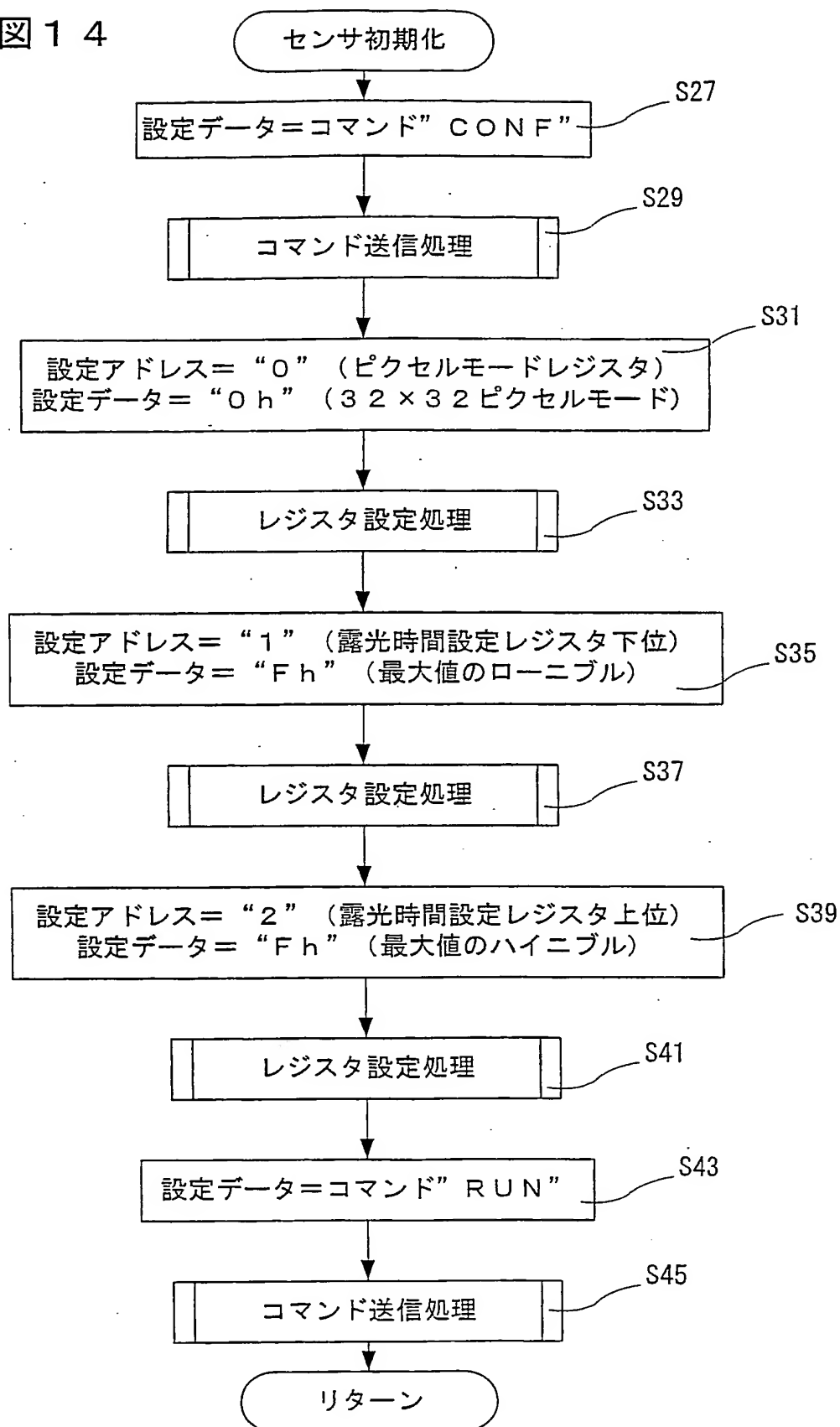


図 1 5

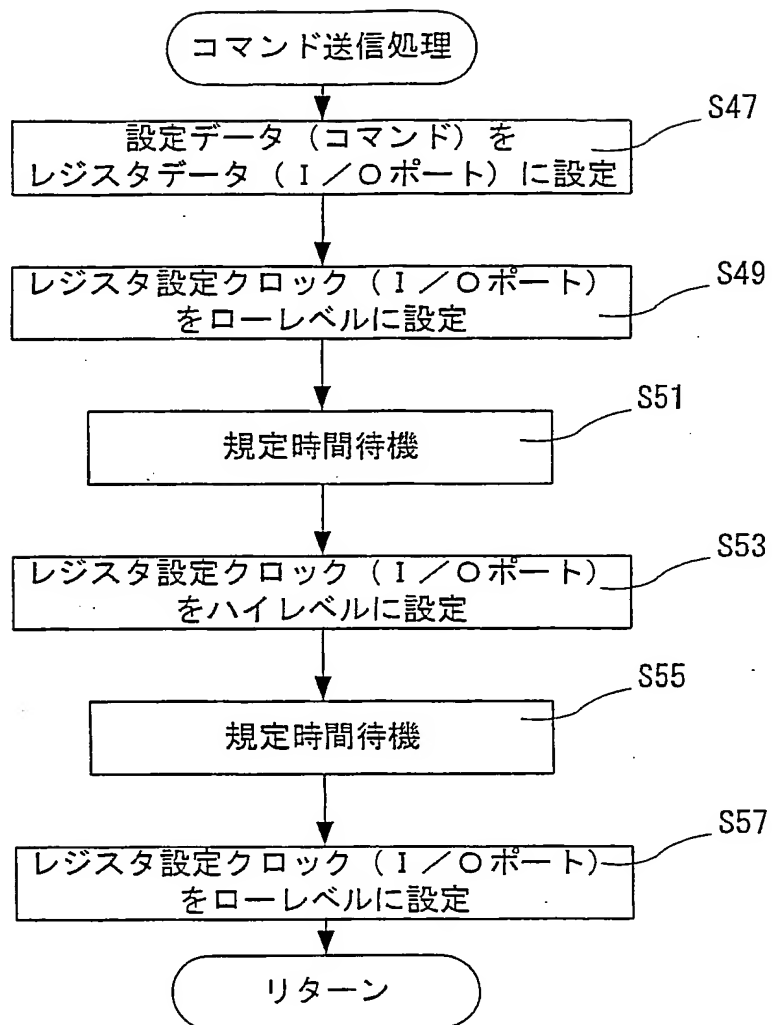


図 16

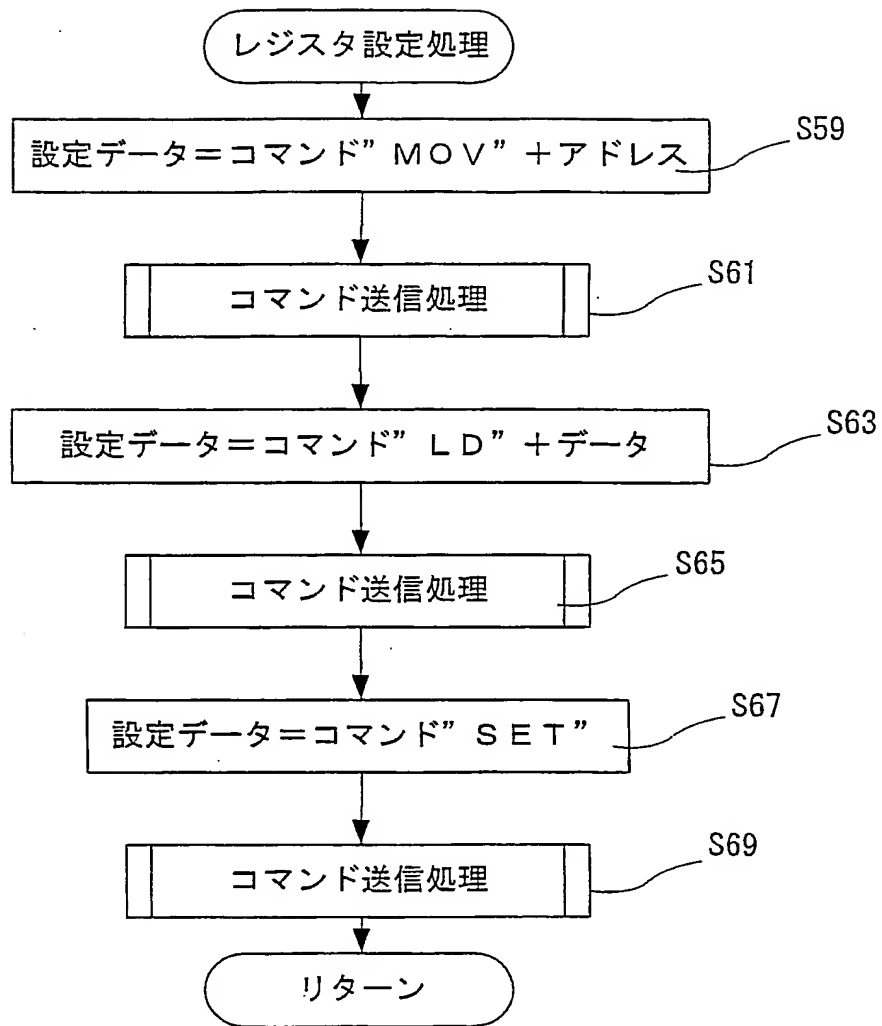


図 17

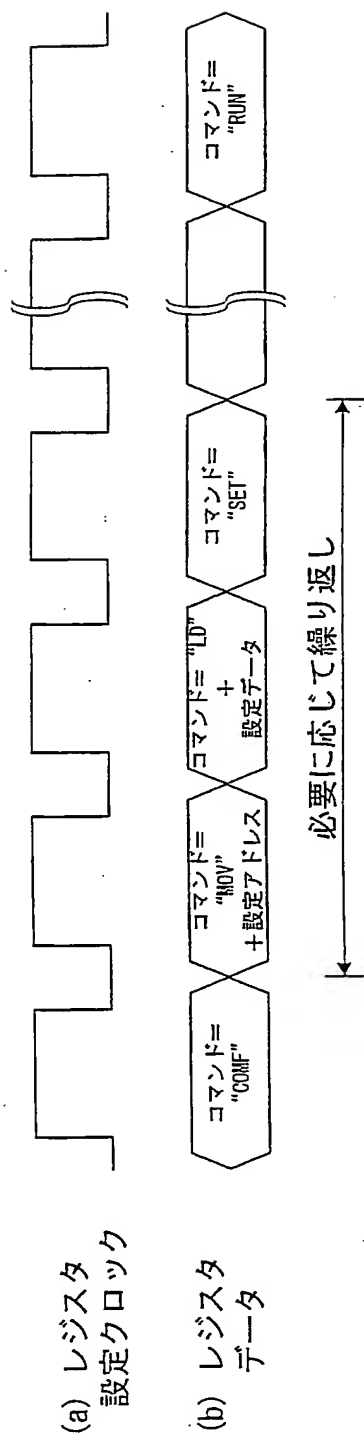


図 18

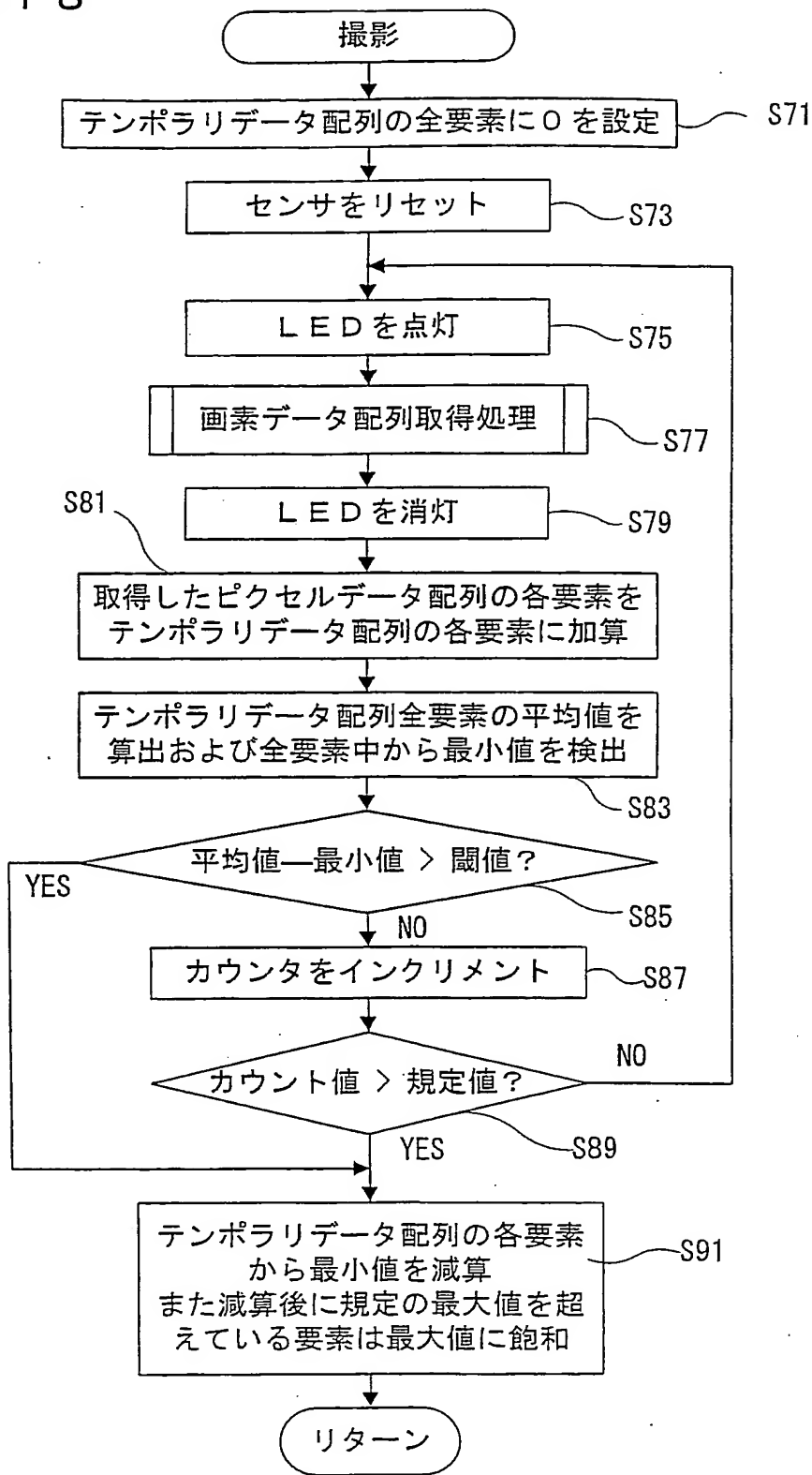


図 19

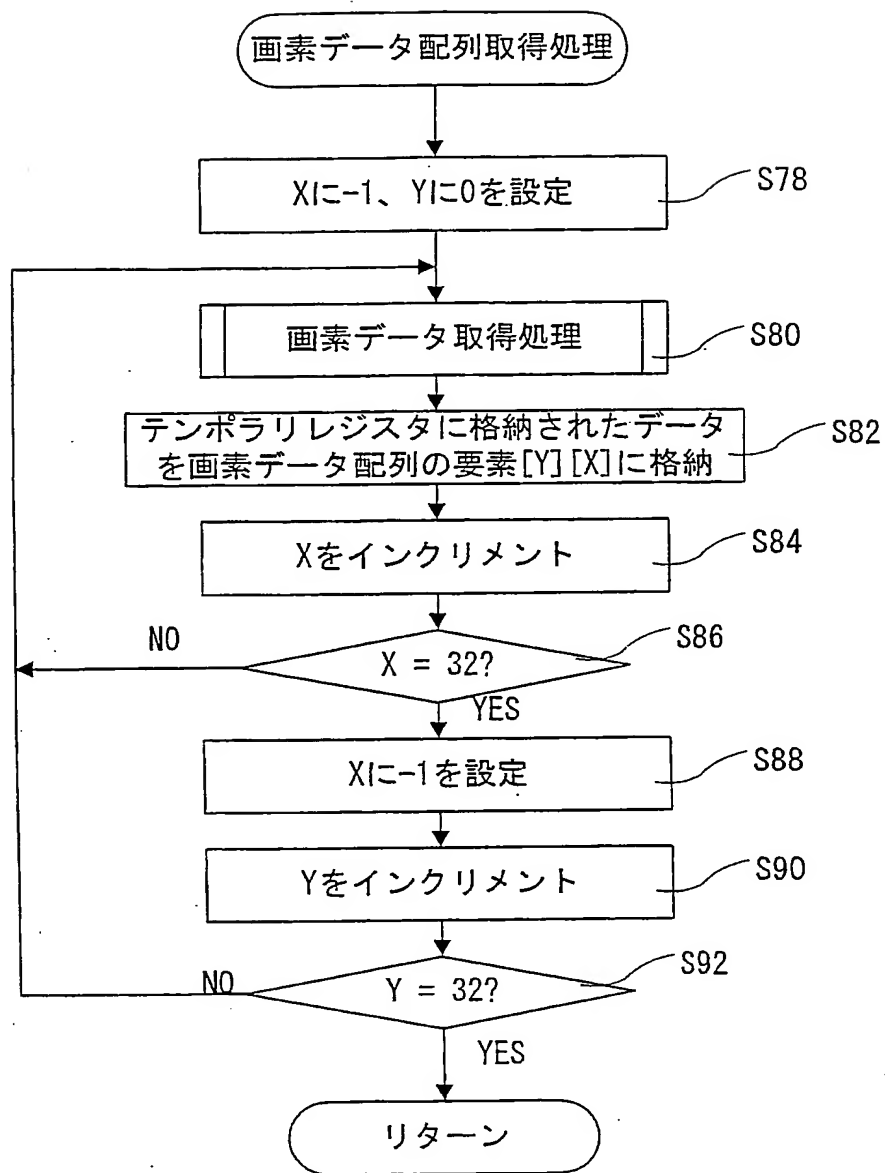


図 20

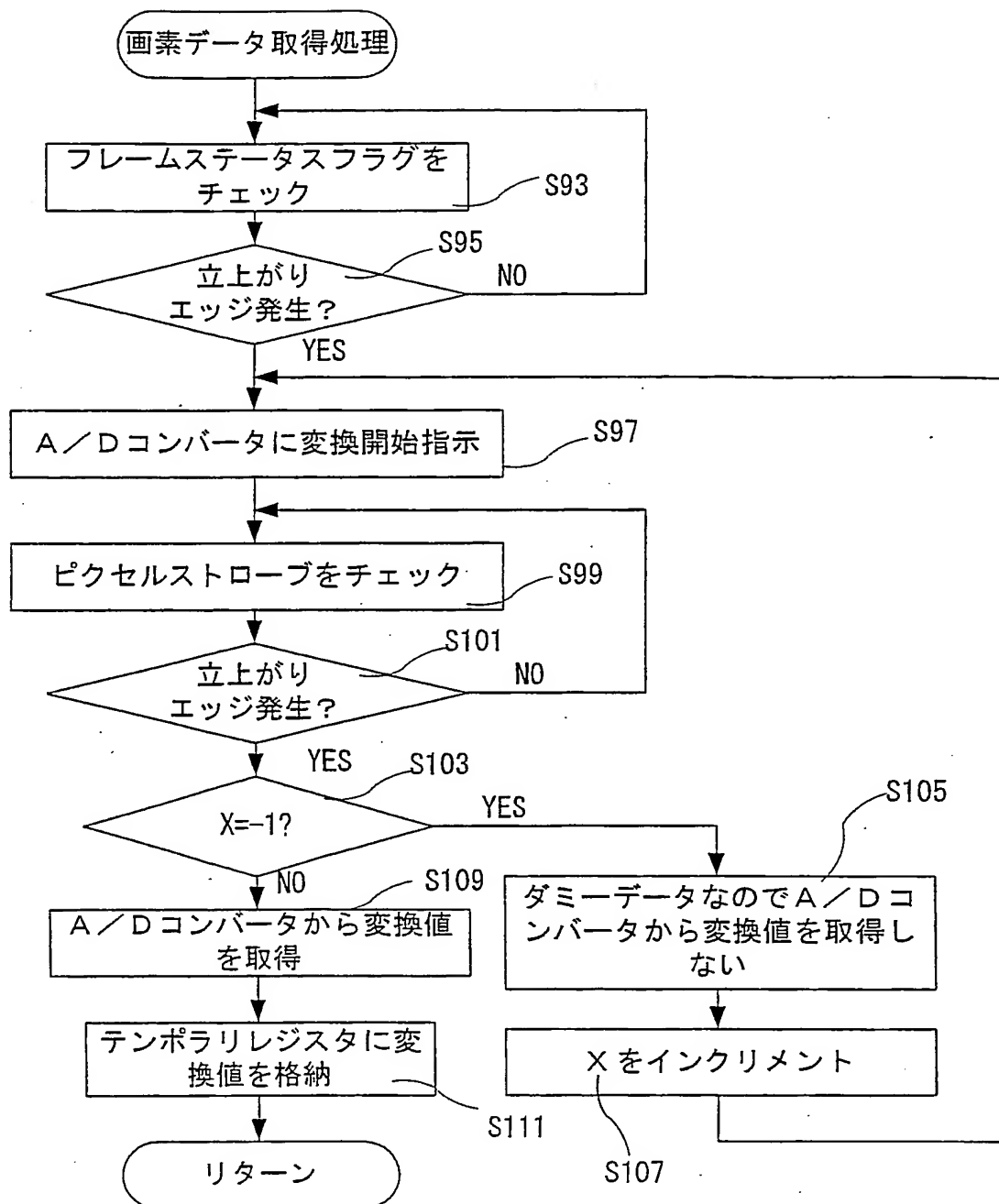


図 2 1

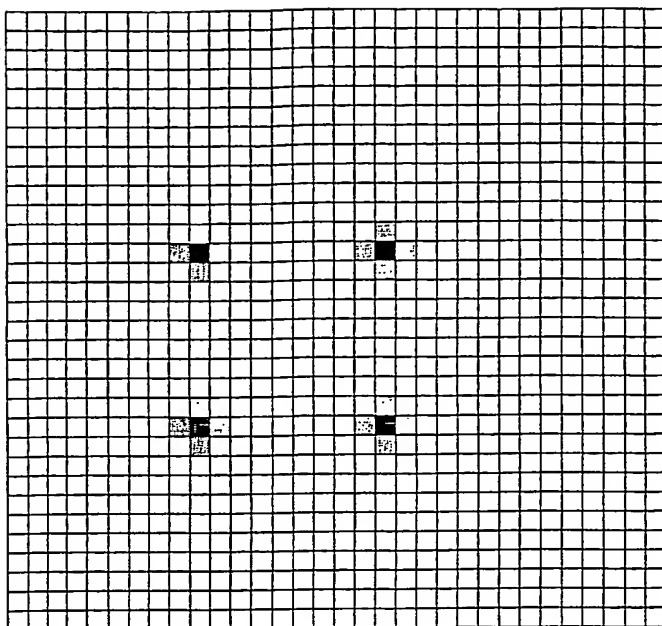


図 2 2

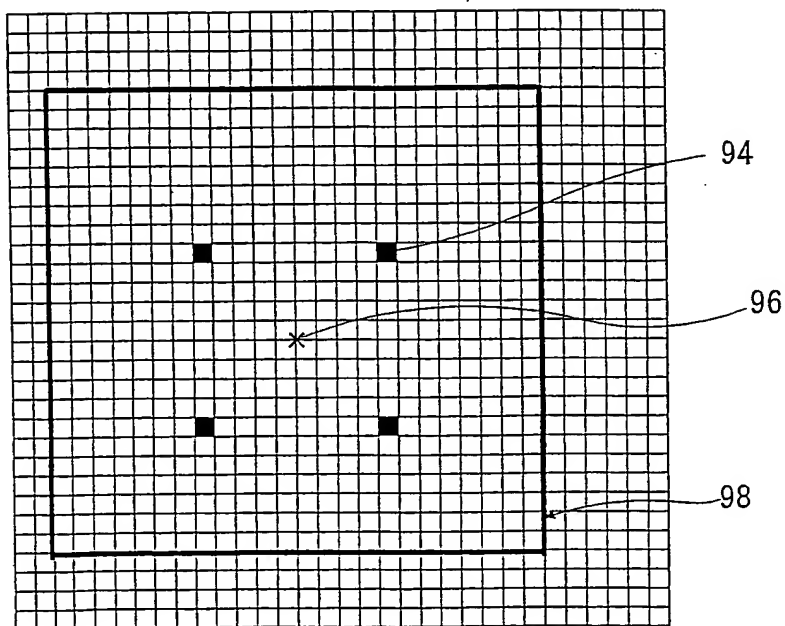


図 2 3

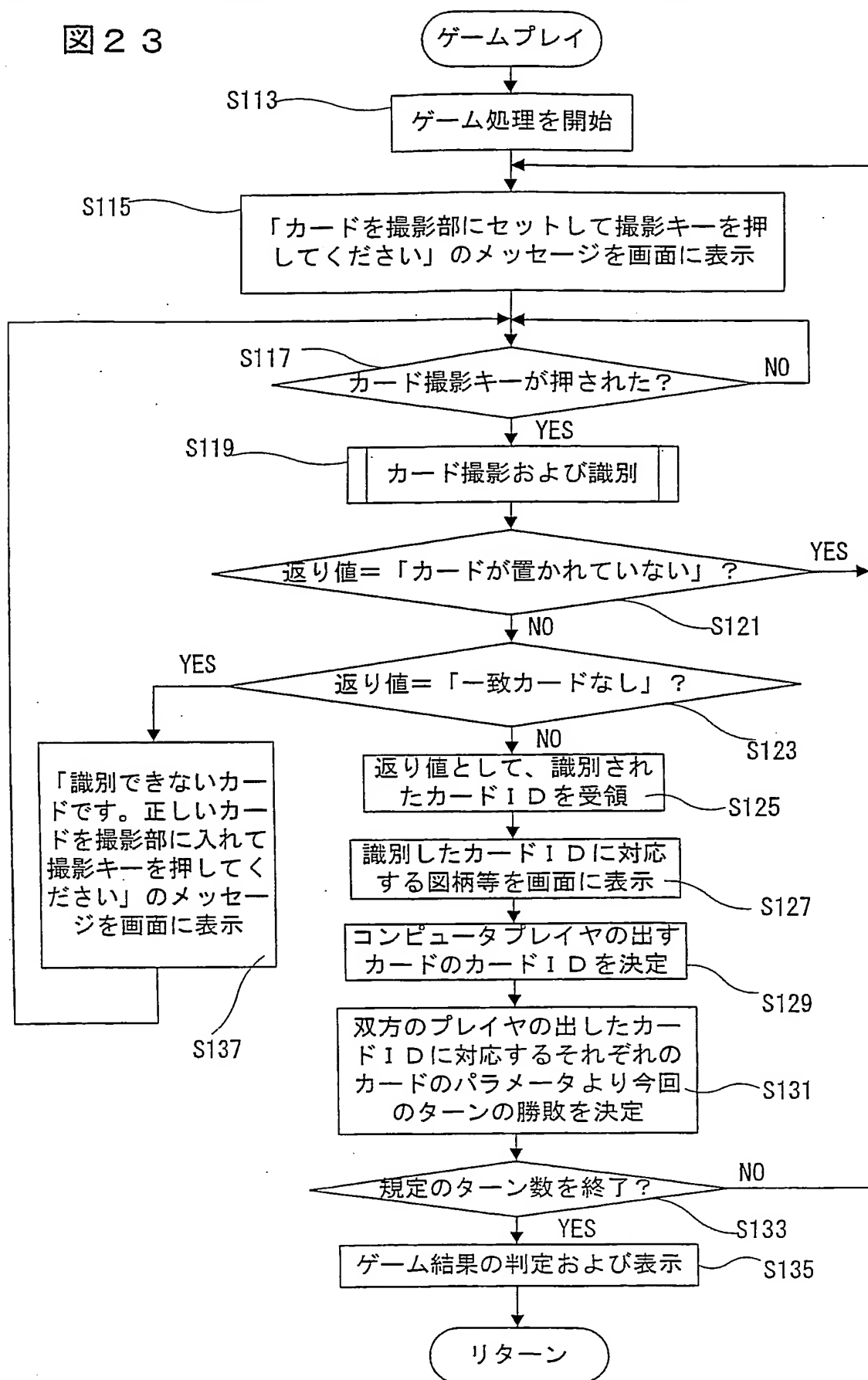


図 2 4

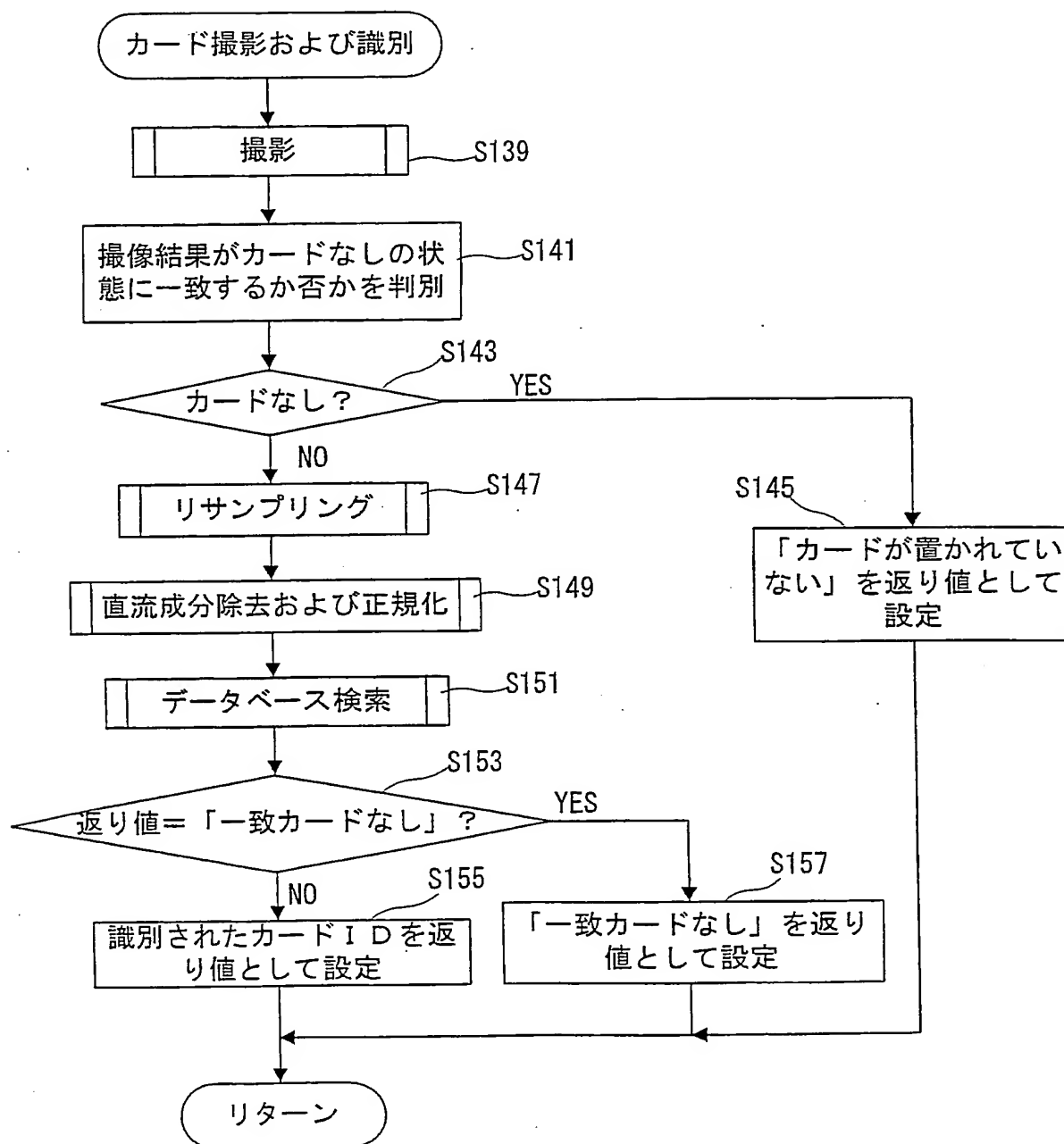


図 2 5

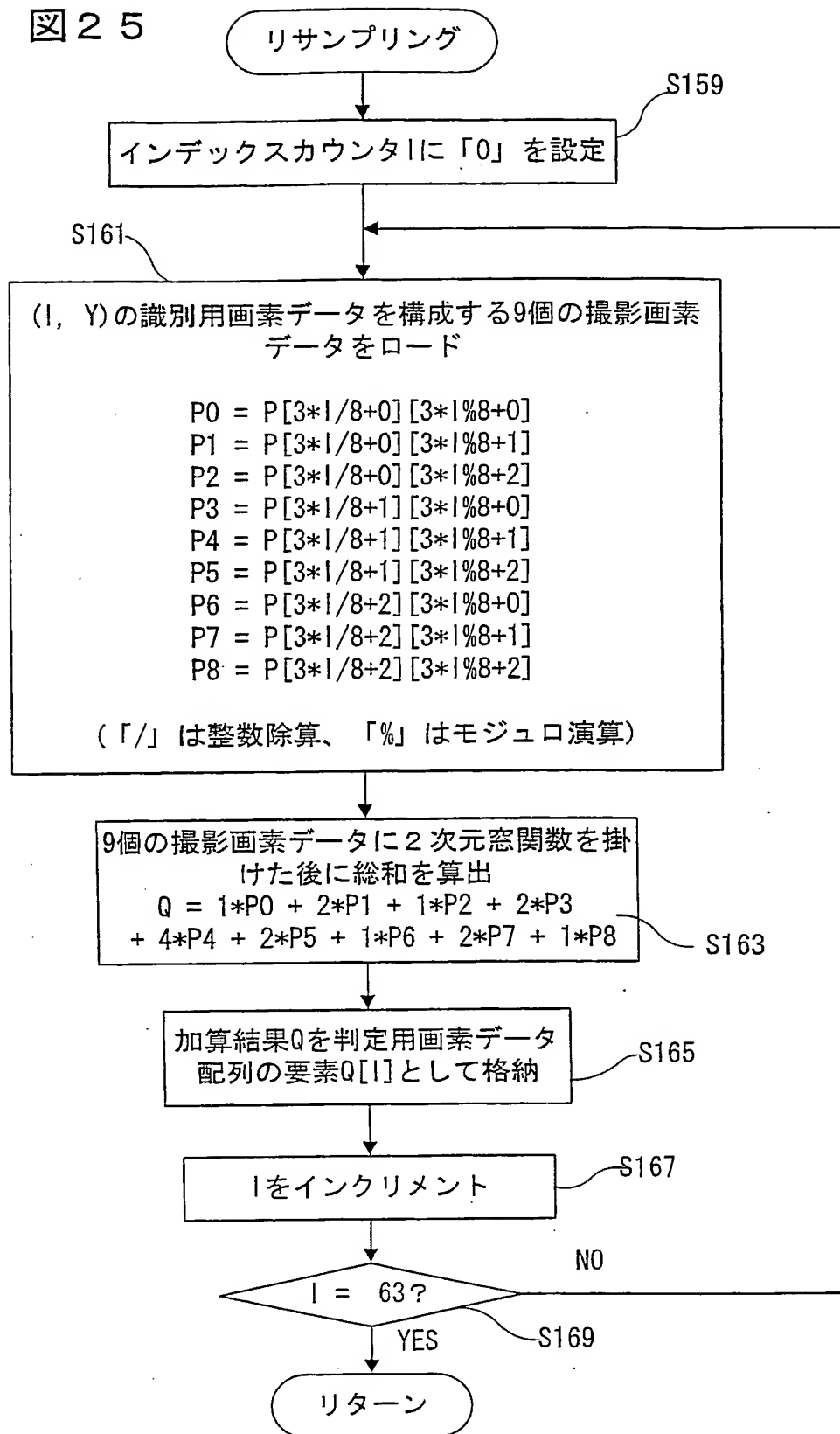


図 2 6

各画素の重み付けの例
(2次元窓関数)

1	2	1
2	4	2
1	2	1

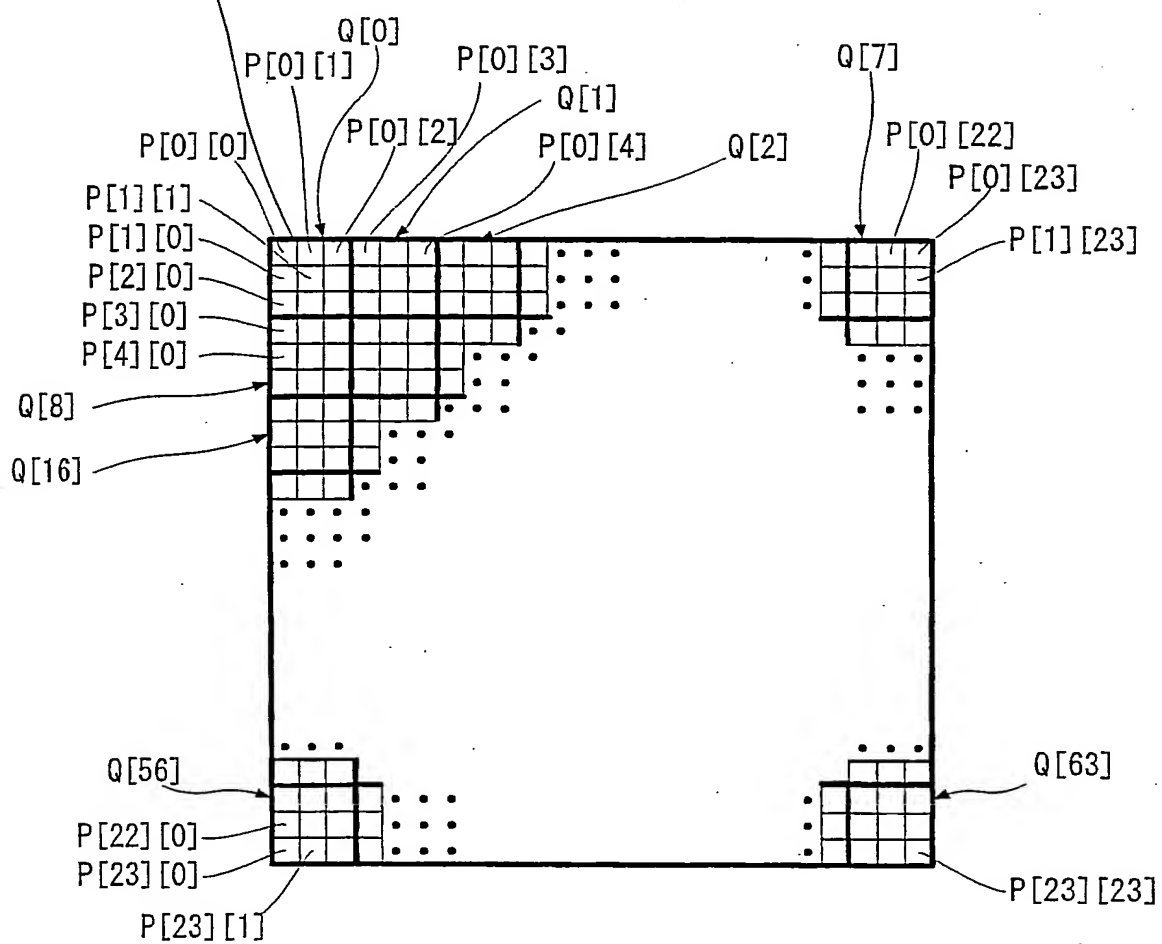


図 27

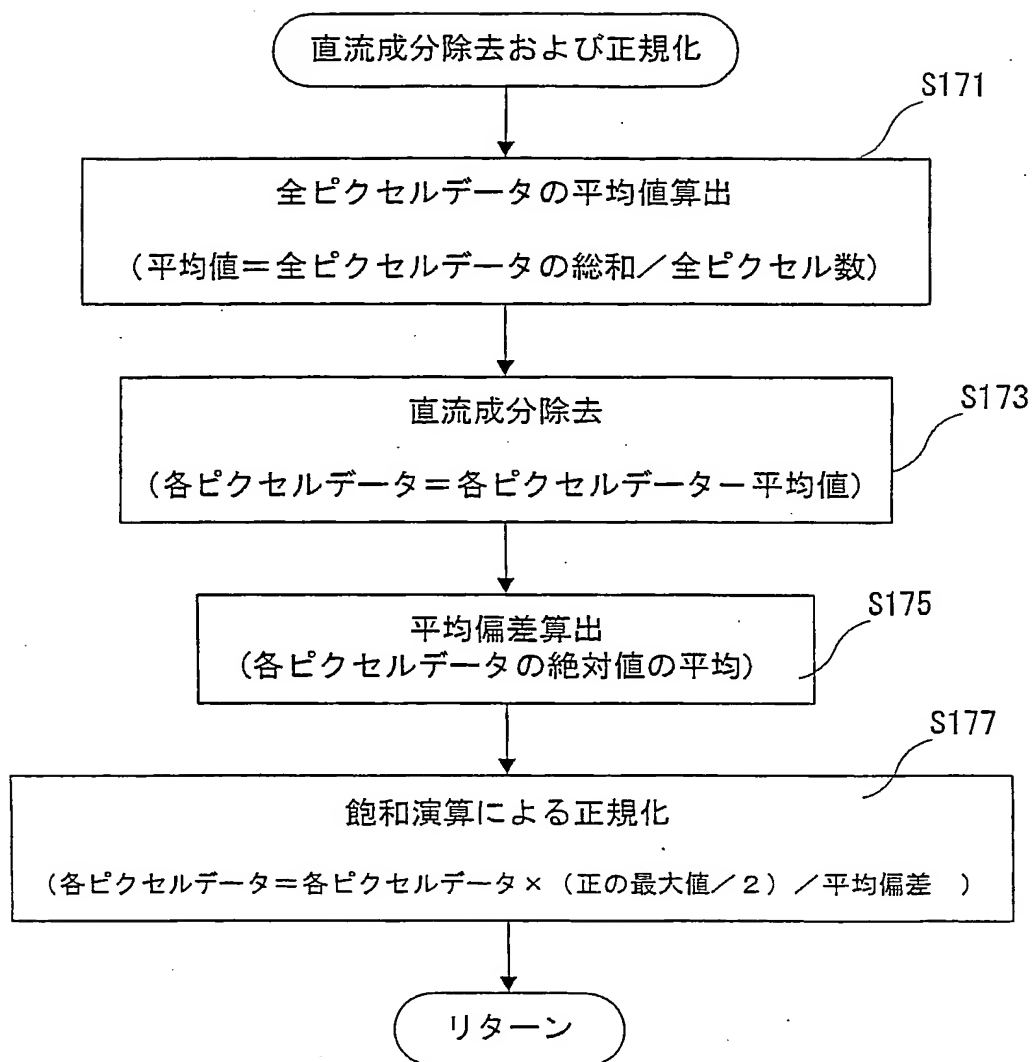


図 28

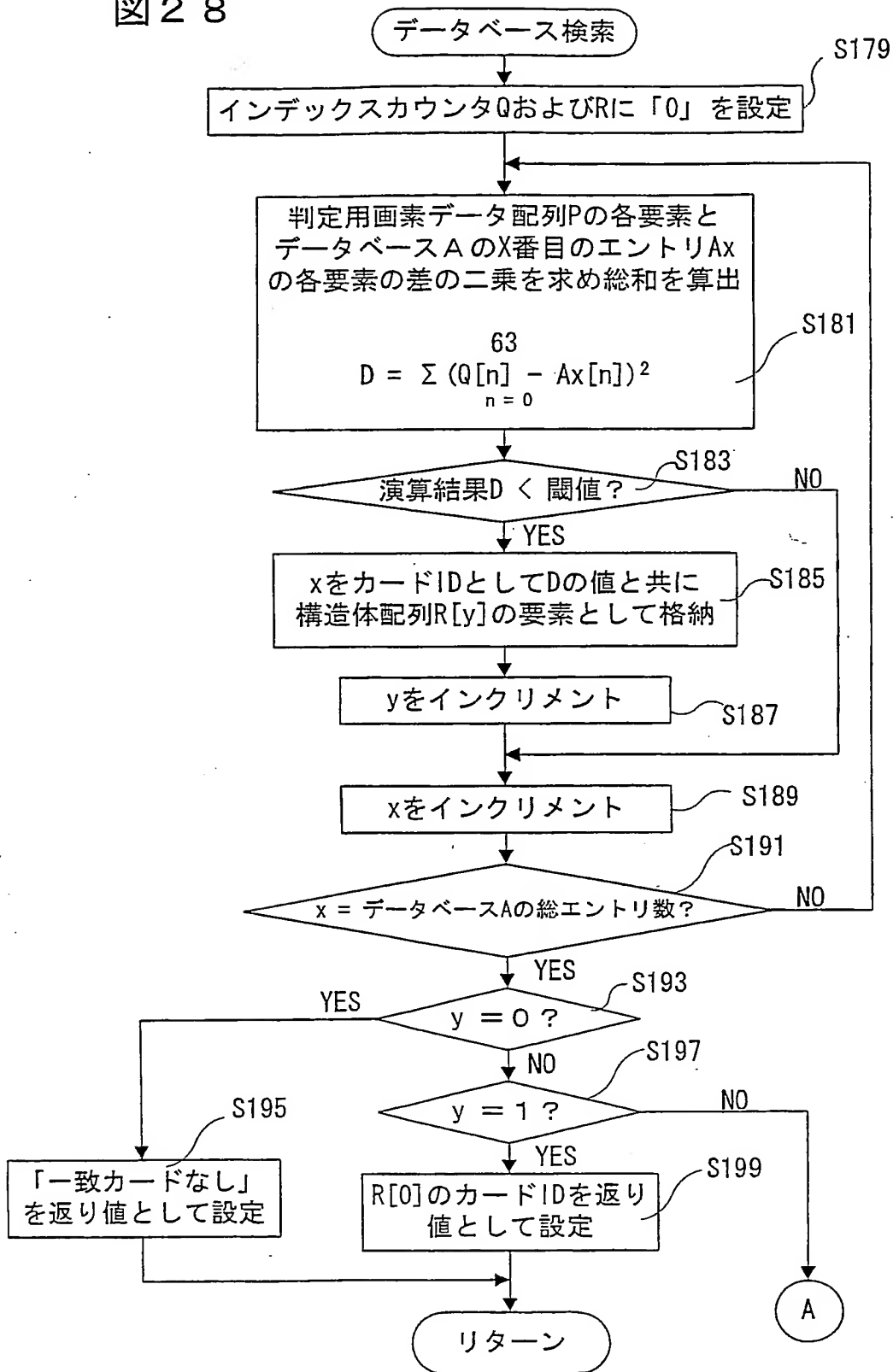


図 29

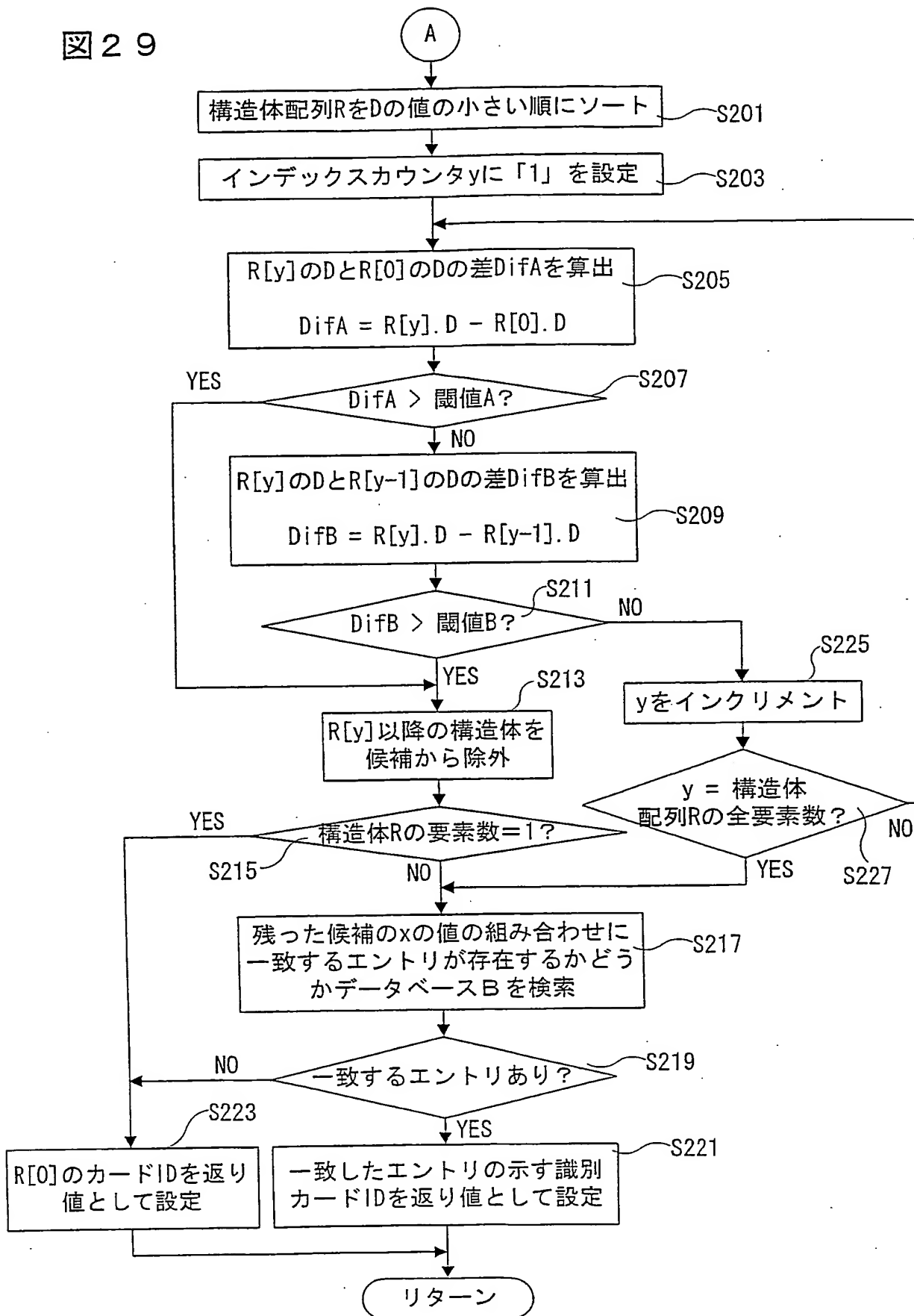


図30

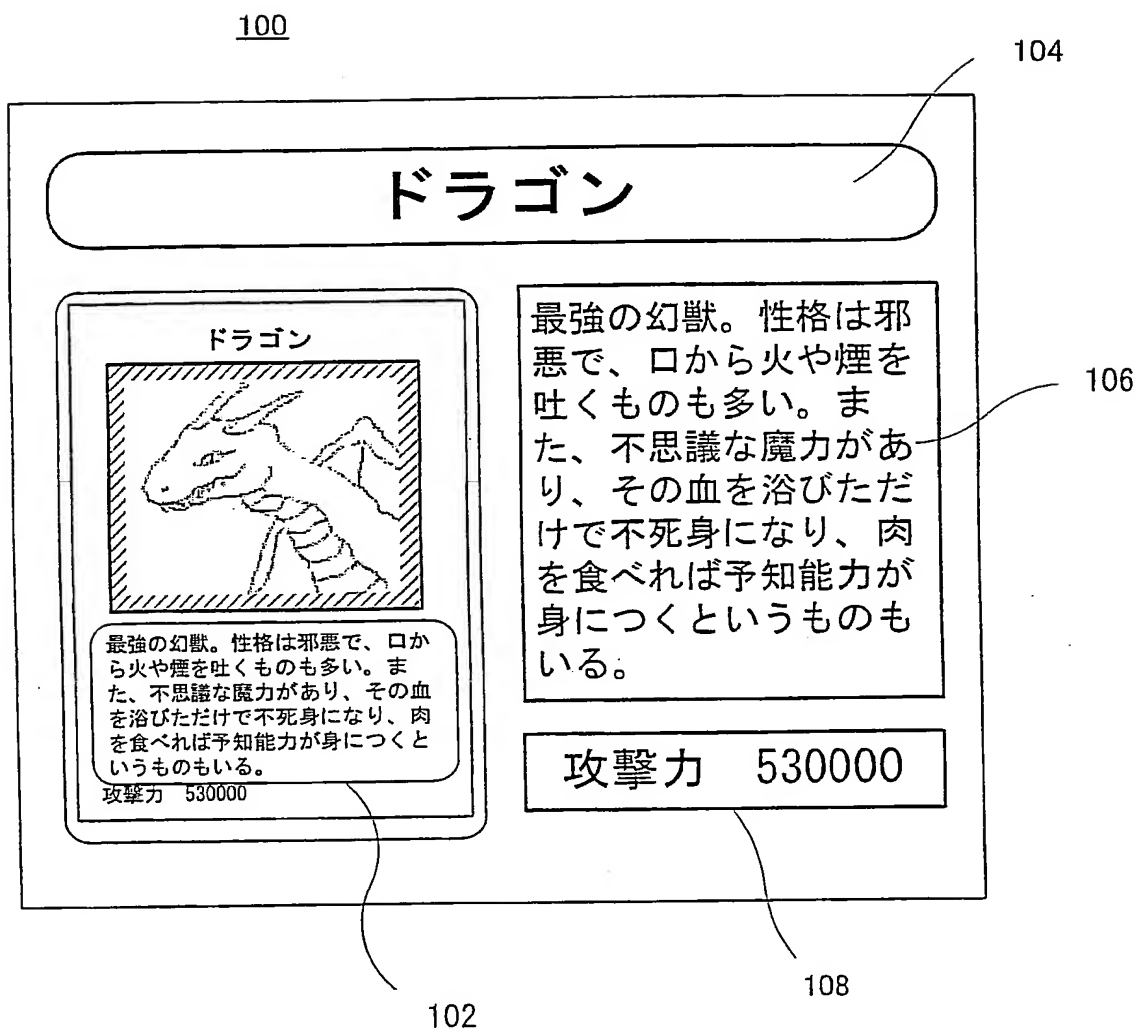


図31

